DTIC® has determined on 06/11/2010 that this Technical Document has the Distribution Statement checked below. The current distribution for this document can be found in the DTIC® Technical Report Database.

☒ **DISTRIBUTION STATEMENT A.** Approved for public release; distribution is unlimited.

☐ **© COPYRIGHTED**; U.S. Government or Federal Rights License. All other rights and uses except those permitted by copyright law are reserved by the copyright owner.

☐ **DISTRIBUTION STATEMENT B.** Distribution authorized to U.S. Government agencies only (fill in reason) (date of determination). Other requests for this document shall be referred to (insert controlling DoD office)

☐ **DISTRIBUTION STATEMENT C.** Distribution authorized to U.S. Government Agencies and their contractors (fill in reason) (date of determination). Other requests for this document shall be referred to (insert controlling DoD office)

☐ **DISTRIBUTION STATEMENT D.** Distribution authorized to the Department of Defense and U.S. DoD contractors only (fill in reason) (date of determination). Other requests shall be referred to (insert controlling DoD office).

☐ **DISTRIBUTION STATEMENT E.** Distribution authorized to DoD Components only (fill in reason) (date of determination). Other requests shall be referred to (insert controlling DoD office).

☐ **DISTRIBUTION STATEMENT F.** Further dissemination only as directed by (inserting controlling DoD office) (date of determination) or higher DoD authority.

Distribution Statement F is also used when a document does not contain a distribution statement and no distribution statement can be determined.

☐ **DISTRIBUTION STATEMENT X.** Distribution authorized to U.S. Government Agencies and private individuals or enterprises eligible to obtain export-controlled technical data in accordance with DoDD 5230.25; (date of determination). DoD Controlling Office is (insert controlling DoD office).

PROCEEDINGS

November 30 - December 2, 1987

ARMY
NAVY
USMC
AIRFORCE
INDUSTRY

9th INTERSERVICE/INDUSTRY TRAINING SYSTEMS CONFERENCE

AMERICAN DEFENSE PREPAREDNESS ASSOCIATION
US

FORWARD from the PROGRAM CHAIRMAN

The American Defense Preparedness Association, in cooperation with approximately 125 industrial companies, Department of Defense agencies, foreign governments, and the academic community, is pleased to host the Ninth Interservice/Industry Training Systems Conference in Washington, D.C., on November 30 to December 2, 1987.

Annually, the Interservice/Industry Training Systems Conference is the single event endorsed by the services in the simulation and training technology arena for the interservice exchange of information among the industry and military training system community. The emphasis has always been, and will continue to be, the free interchange of information related to simulation and training.

Year in and year out, papers presented and published at the conference have served as the primary vehicles for this *"free interchange of information"*.

The theme for the 1987 conference,***"TRAINING SYSTEMS-THE CRITICAL ADVANTAGE"***, focuses attention on the use of training systems as the primary means of ensuring cost effective operational readiness. As the sophistication of advanced weaponry and support systems has increased, so has the criticality of the training provided to the individuals and units which use and maintain that equipment. This evolution of systems has provided the impetus for development of new and innovative ways to train. These new methodologies must now be employed to guarantee optimum skill levels among the operators and maintainers of our defense systems. Appropriately applied, *training systems can be the critical advantage.*

The Army, Navy, Air Force and Marine Corps have steadily evolved toward a total systems philosophy in the acquisition of training equipment. As this evolution proceeds, it is apparent that training systems must be planned and developed from the Manpower, Personnel, and Training (MPT) viewpoint concurrently with the weapon system. As the major conference of its type, the I/ITSC plans to ensure that training systems and MPT requirements are emphasized to be responsive to the services and industry.

Communication between the military and industry is *critical* to the advancements in training systems. Without a clear understanding of the military's needs and problems in training, effective solutions are doubtful. Conversely, military planners cannot capitalize on the technical and management advances of industry unless they are communicated. The goal of the Interservice/Industry Training Systems Conference is to serve as a forum to foster this communication. Through technical, management, and user papers, panels, exhibits, and individual discussions, problems are aired, needs are defined, and plans are specified to develop alternatives and solutions.The purpose of this conference is this communication and the papers contained in these proceedings are an attempt to document both industry's and government's views.

Enormous credit for putting together a conference of this magnitude must go to the many individuals who have contributed their talents and time so generously and to the organizations that supported them. These people are identified on the following pages. A special word of thanks to my assistant, Marylou Gow, for her tireless efforts in preparation of these proceedings.

Arthur L. Banman
Program Chairman

INTERSERVICE EXECUTIVE COMMITTEE

COLONEL RICHARD J. LUNSFORD, JR. United States Army
Project Manager for Training Devices, Orlando, Florida
CAPTAIN CHARLES DANA ROWLEY, United States Navy, Commanding Officer
Naval Training Systems Center, Orlando, Florida
COLONEL PETER R. ROUNSEVILLE, United States Marine Corps
Marine Corps Liaison Officer
Naval Training Systems Center, Orlando, Florida
COLONEL WAYNE J. LOBBESTAEL, United States Air Force
Deputy Commander for Training Systems
Wright Patterson AFB, Ohio

INTERSERVICE STEERING COMMITTEE

Ralph C. Nelson, Chief, Program Management and Requirements Branch
PM TRADE, United States Army
James M. Clemens, Jr., Head, Support Systems Engineering Division
Naval Training Systems Center
LtCol John Mills, USMC, V-22 Project Manager
Naval Training Systems Center
Udell Albrechtsen, Chief, Logistics Management Branch
Ogden Air Logistics Center, United States Air Force
F. J. Winter, Jr., Director, Test & Deployment, Deputy for Training Systems
Aeronautical Systems Division, United States Air Force

CONFERENCE COMMITTEE

CONFERENCE CHAIRMAN, David P. Crane, Rediffusion Simulation Inc.

PROGRAM CHAIRMAN, Arthur L. Banman, Gould Inc., CSD

ADMINISTRATIVE ASSISTANT, Marylou Gow, Gould Inc., CSD

SPEAKER CHAIRMAN, Lt Col Robert Croach, USAF, OSD Training Policy

PUBLICITY CHAIRMAN, A. Q. Collier, Naval Training Systems Center

CONGRESSIONAL LIAISON CHAIRMAN, R. Q. Old, R. Q. Old Associates, Inc.

TECHNICAL SUBCOMMITTEE, Ralph T. Davis, Phoenix International

MANAGEMENT SUBCOMMITTEE, Sam C. Worrell, Air Analytics

USER SUBCOMMITTEE, Don Campbell, DAEDALEAN Inc.

MPT LIAISON, COL(P) Jack A. Pellicci, USA, MANPRINT Policy Office

BEST PAPER SUBCOMMITTEE, Phillip Sprinkle, PM TRADE

ACADEMIA LIAISON, Dr. Christopher Bauer, University of Central Florida

ADPA REPRESENTATIVES,
CAPT Nelson P. Jackson, USN (Ret.), Assistant Director, Advisory Service
John A. Todd, The Singer Company, Chairman, Training Systems Division, ADPA

TECHNICAL SUBCOMMITTEE

Chairman
Ralph T. Davis, Phoenix International

Members
Lt Col Dave Irwin,USAF, Deputy for Training Systems
Richard O'Dell, Deputy for Training Systems
Edward Boyle, AFHRL/LRC
Dr. Leonard Courchesne, NAVAIRSYSCOM
William Rizzo, NAVTRASYSCEN
Donald Jones, NAVTRASYSCEN
Dr. Tom Gray, AFHRL/OTU
Oliver Nelson, HQ ATC/TTST
Dr. Frank Schufletowski, HQ ATC/TTXA
Jack Ackerson, Deputy for Training Systems
Ken Steffee, NAVAIRSYSCOM
Joe Cianfrani, NAVAIRSYSCOM
Thomas Mazza, PM TRADE
SMSgt Terry Heffelfinger, USAFTAWC
Lt Col Dave Pohlman, USAF, AMD/RDST
Hugh Merry, Ogden Air Logistics Center
Maj Don Bailey, HQ USMC
A.E. Andrews, Los Alamos National Lab
Gene Brown, McDonnell Douglas
Bob Schwing, The Singer Company
Dorothy Baldwin, General Dynamics
James Burke, Gould CSD
Lynn Thompson, MDHC
Dr. Ron Hughes, MDHC
James R. Cooley, AAI
Jon Casperson, BMAC
Matt Narotam, Burtek
Ralph Hileman, DRS
John Gray, Honeywell, Inc.
Dr. Bruce McDonald, Harris Corp.
Dr. Murdoch McKinnon, CAE Electronics Ltd.
Dr. Norman R. Potter, Applied Science Associates Inc.
Jeff Pulcini, Concurrent Computer Corp.
Gary Shrock, Cubic Corp.
Frederic W. Snyder, Boeing Military Airplane Co.
George C. Snyder, Goodyear Aerospace Corp.
Robert Pippin, LTV Missiles & Electronics Group
Dr. Robert Bloom, G.P. Taurio
Gareth Beitzel, HQ ATC/TTST
Hal Geltmacher, AFHRL/OT
John Hammond, AAI Corp.
Al Irwin, SAIC
Dr. Stanley Levine, Douglas Aircraft
John Marrs, National Training Center
Greco Myren, Honeywell Inc.
Dr. Renata Schmidt, General Physics Corp.
Dr. Lois Wilson, Ford Aerospace

MANAGEMENT SUBCOMMITTEE

Chairman
Sam Worrell, Air Analytics

Members
George Dickison, Deputy for Training Systems
Karl Jackson, PM TRADE
Steven McKee, TPDC
Ken Potempa, HSD XR
Tim Sharp, Deputy for Training Systems
Dr. Mike Zajkowski, NAVTRASYSCEN
Mike McGaugh, United Airlines Services Corp.
Dr. Jane Wager, NAVTRASYSCEN
Lt Col Jim Wansack, USAF, ASAFF/AQPN
Kenneth B. Beckham, Reflectone Inc.
Larry J. Gallagher, Loral Electro-Optical Systems, Inc.
Robert Goforth, Southwest Research Institute
Gene Holcomb, Honeywell Inc.
James A. Horton, Loral Systems Group
Chuck Monachello, Singer Link
Ann O'Kennon-Zimmerman, EER Systems
Bo Pogoloff, AAI Corporation
Ed Porterfield, The Singer Co. (LSSD)
Judith Shellnut-Riess, Syscon Corp.
Jerry Shimp, Information Spectrum, Inc.
Wayne I. (Skip) Yohe, McDonnell Aircraft Company
Dennis Shockley, Boeing
Dr. Ruth Wienclaw, Wienclaw Associates
Jack Drewett, General Dynamics
James Mandaracchia, SAIC
Frank Lawler, SYNTEC
Jerry Balz, Logicon
Marion D'Argenzio, DYNALECTRON
Neil Camp, McDonnell Douglas
Jerry W. Jerome, Ball Corp. VERAC
Rob Rhode, DEC
Eric Monroe, MDHC
James Nelson, LTV
MGySgt R. D. Leavitt, HQMC(T)
Lt Col Eugene Clayton, USAF, HQ AFSC/SDTA
Lt Col Ronald Dukes, USAF, HQ MAC/DOT
Jay Irwin, NAVAIRSYSCOM
CDR Cliff Hathway, USN, OPNAV(OP-593)
Glenn Hohman, NAVTRASYSCEN
Eugene Lease, NAVTRASYSCEN
Col Michael Lane, USAF, AFHRL/OT
James O'Connell, The Singer Company
Stephen Sheppard, NAVSEASYSCOM
Col Christopher Somers, USAF, ASD/ALH
Ms. Jo D. Voeller, Ogden Air Logistics Center

USER COMMITTEE

Chairman
Don Campbell, DAEDALEAN Inc.

Members
Col Ron Baker, USAF, OMC/CC
Jim Bishop, NAVTRASYSCEN
Maj Andy Courtice, USAF, Deputy for Training Systems
Hal Hansen, US Army Training Support Center
Billy Kennedy, US Army Forces Command
Maj Andrew Broadstone, TAGSC MCDEC
CDR Chuck Sammons, USN, NAVAIRSYSCOM
C. J. "Bud " Chappell, NAVTRASYSCEN
Larry Barks, NAVAIRSYSCOM
Dr. Dee Andrews, ARI
Col Joseph Crozier, USAFTAWC/TM
Dr. Gerry Albers, General Electric Co.
Neale Cosby, Perceptronics Inc.
Frank Hines, PSE
Mike Monaghan, Reflectone Inc.
Larry Vernon, Evans & Sutherland
Pres McGee, The Singer Company
Bob Stevenson, RSI
J. H. Herbert, General Dynamics
Jim Cooksey, Advanced Technology
Steve Selcho, Evans & Sutherland
Gary Baker, Boeing
Jack Nicholson, Consultant
Paul Bernhardt, Consultant
Lt Col Bob Boswell, USAF, HQ SAC-SES CC
Richard Dickson, SAIC
Lt Col Larry Gosnell, USAF, HQ SAC/XPHN
COL George Harmeyer, USA, DAMO/TRS
James Jackson, LTV
Jeff Marlin, Firearms Training Systems
Capt Brian O'Malley, USAF, HQ AFSPACSCOM-DOT
Dr. Judy O'Neil, Dynamics Research Corp.
Col Dennis Smith, HQ USAF AF/DPPT
Maj Ron Thompson, USAF, HQ MAC/XPQS
Lt Col Raymond Wilcox, USAF, HQ TAC/DOTS
Albert C. Zimmerman, US Army, ATSC
James Slate, Burtek

TABLE OF CONTENTS

TECHNICAL PAPERS **PAGE**

TABLE OF CONTENTS(continued)

TECHNICAL PAPERS **PAGE**

TABLE OF CONTENTS(continued)

TABLE OF CONTENTS(continued)

TABLE OF CONTENTS(continued)

TECHNICAL PAPERS **PAGE**

TABLE OF CONTENTS(continued)

TECHNICAL PAPERS **PAGE**

TECHNICAL PAPERS FOR PUBLICATION

TABLE OF CONTENTS(continued)

MANAGEMENT PAPERS **PAGE**

MANPOWER, PERSONNEL & TRAINING

INTEGRATED LOGISTICS SUPPORT

ACQUISITION STRATEGIES I

TABLE OF CONTENTS(continued)

MANAGEMENT PAPERS **PAGE**

ACQUISITION STRATEGIES II

MANAGEMENT INITIATIVES

MANAGEMENT PAPERS FOR PUBLICATION

TABLE OF CONTENTS(continued)

USER PAPERS **PAGE**

COLLECTIVE TRAINING

CONTRACTOR'S ROLE IN TRAINING

TRAINING STRATEGIES

TABLE OF CONTENTS(continued)

USER PAPERS **PAGE**

TRAINING DEVELOPMENTS

COMPUTER-BASED TRAINING

USER PAPERS FOR PUBLICATION

MANPOWER, TRAINING, AND DOCUMENTATION ANALYSES: NOT STRANGE BEDFELLOWS

Robert S. Bloom, Renata V. Schmidt, and William P. Hardy
GP Taurio, Inc./General Physics Corporation
1901 No. Beauregard St., Suite 300
Alexandria, VA 22311

ABSTRACT

This paper specifies a conceptual model for training system development describing the interrelationship of MPT Resource Requirements Analysis, ISD training content development, and technical documentation for military tactical weapon systems and training devices. It describes how a common data base containing specific performance data drives a variety of analytic, resource determination, and requirements decision tasks. It discusses the interface points and impacts of the three military training system development components on each other and on their products. It demonstrates how military and contractor manpower and training analysts and hardware and software engineers can coordinate their data collection, analysis, and documentation efforts in a timely and cost-efficient manner.

INTRODUCTION

The use of training systems as a means of ensuring cost-effective readiness in the military arena has focused attention on three areas of analysis to derive relevant and appropriate training decisions. The mandated front-end analysis of Manpower, Personnel, and Training (MPT) and Human Factors Requirements (including safety and hazards) establishes resource and human impacts on weapon system design and life cycle support issues (e.g., the Navy HARDMAN, Army MANPRINT, and Air Force front-end analysis approaches). Development and use of a Systems Approach to Training (SAT) embodied in the Instructional Systems Development (ISD) model (MIL-STD-001379(C) Navy), MIL-T-29053, NAVSEA OD 45519, AF Pamphlet 50-58 and AF Manual 50-2, and others) grew from concerns about training approaches and content provided to military personnel who operate, maintain, instruct, supervise, or support weapon systems/subsystems/equipment. The third area considers technical documentation of weapon system operation and maintenance as it relates to the equipment and its use in providing training support for new trainees and trained personnel whose skills need to be upgraded or maintained.

These three masters--MPT front-end analysis, ISD requirements for training content development, and technical documentation--are all served by a variety of data-gathering tasks related to equipment or systems performance. Analysts use the data to define the quantity of manpower; the quality of personnel; the breadth, scope, and nature of the training program; the kinds of training strategies and training devices employed; and the structure of technical documentation. The commonality of the data required to perform analysis and make requirements decisions, and the similiarity of the analytic approaches, however, present an opportunity for a Military Training Systems Integration Model that suggests a more cost-efficient and effective data-gathering methodology. Our objective in this paper is to describe an integrated approach for training systems development so that MPT analysts, training analysts, instructional technologists, technical writers, and hardware/software engineers can create and use data needed for their respective analytical, developmental, and decision roles, and interact with each other to minimize data duplication and enhance data coordination.

This paper introduces a conceptual model for the purpose of training system development showing the interrelationship of MPT Resource Requirements Analysis, ISD applications, and technical documentation requirements for tactical weapon systems and training devices. The model supports an integrated systems approach to the collection, analysis, and production of data for making resource requirements, curricula and content, and technical documentation decisions as acquisition of weapon systems/training devices progresses from concept exploration through life-cycle implementation. By displaying the approaches to MPT, ISD, and technical documentation practices side by side, the model identifies points of interface germane to a systematic and conceptual interplay, and demonstrates how a common data base (quantitative and qualitative statements of functional performance, and the results of needs, population, and constraints analyses) serves as the foundation for analysis and subsequent decision making in military training. Further, the model provides a structure for discussion of critical training questions (what, why, how, where, when, how many, how good, etc.). We do not intend to discuss all indicated steps in the model but will concentrate on interfaces where the model suggests a way to enhance or optimize efforts to develop MPT, ISD, and technical documentation products.

Training Systems Integration Model

The Training Systems Integration Model, shown in Figure 1, begins with a common data base. Initial performance data are gathered so they are available to MPT, instructional, and engineering analysts. These include (1) personnel tasks or duties necessary to the operation, maintenance, or support of a weapon system; (2) functional specifications of equipment, which present the breadth and scope of what the system is going to do; (3) equipment performance goals and standards specified in terms of quantitative and qualitative opera-

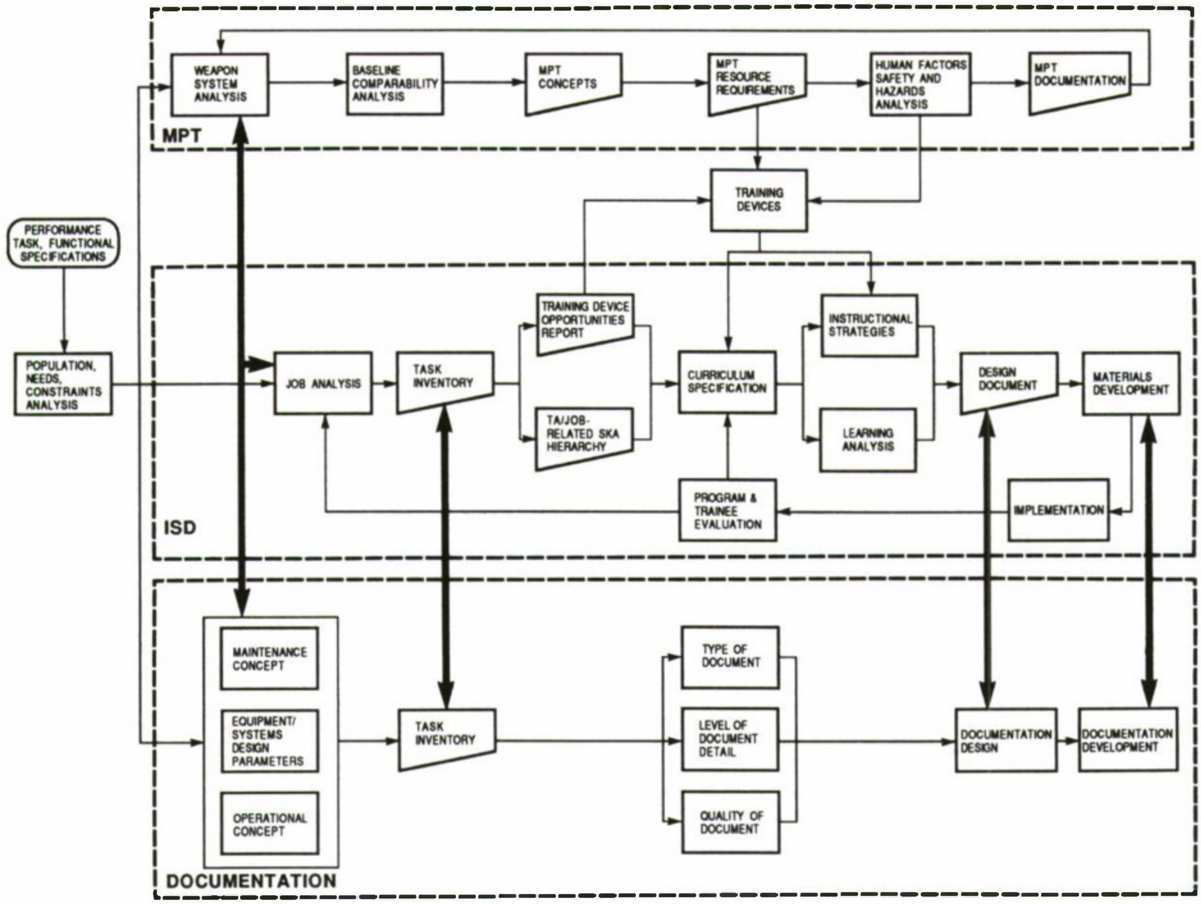


Figure 1. TRAINING SYSTEMS INTEGRATION MODEL

tional and maintenance parameters within which the system will operate, and (4) data on the population, its training and programmatic needs, and various environmental, budgetary, and other constraints. MPT analysis develops estimates of the required MPT resources and human factors impacts to answer weapon system/ training device design questions, to resolve supportability issues regarding the quality and quantity of the manpower pool, and to determine the life-cycle cost of the design, development, and deployment of the system. ISD utilizes similar initial data to establish job and task lists, performance profiles, training objectives, level of content detail (theory, familiarization, practice), training program definition, materials or devices to facilitate trainee achievement of learning objectives (including manuals, training software, etc.), and trainee and program evaluation mechanisms. For technical documentation, the operational and maintenance concepts are solidified and equipment design finalized, establishing the framework for the structure and scope of system technical manuals and operation/maintenance procedures. Despite their differences, each of these areas benefits from critical interfaces beyond the initial use of common data.

MPT Analysis

Manpower, Personnel, and Training Resource Requirements Analysis is a front-end process that uses a systems definition and baseline comparison approach identifying and comparing the system requirements of a new tactical system with a predecessor. Performance data, including workload and the frequency and duration deltas of personnel tasks and functions, serve as the basis for making manpower quantity, personnel quality, and training resource determinations for the new system. In addition, the training devices and approaches used to prepare manpower in the baseline or predecessor system are identified so that determinations about any new devices can be made.

MPT analysis yields decisions concerning the training concept and resource requirements for the new system, which affect (and are, in turn, affected by) ISD and technical documentation development. The point of interface with ISD relates, in part, to the kind of training device to be considered (simulator, piece-part, etc.), the trainee goals it will need to meet, cost factors for initial and life-cycle implementation, and the hardware and software speci-

fications for the training device in relation to the tactical equipment. MPT analysis for a new tactical system also has an impact upon the operational, maintenance, training, and other concepts developed in the system's early technical documentation.

MPT Resource Requirements Analysis is an iterative process, and both the Navy HARDMAN and Army MANPRINT methodologies emphasize the impact of MPT factors on weapon system design, and the effect of initial weapon system design on issues of manpower supportability within the total military force and on manpower operations and maintenance issues (e.g., safety and hazards potentials, human factors, and cognitive issues). Once design decisions have been made, and the manpower quantity and personnel quality have been determined, MPT iterations provide relevant information for the ISD process and for development of technical documentation of system operation and maintenance requirements. In turn, the MPT analysis is affected by both ISD and technical documentation data. Training resource issues, e.g., identifying training facilities, classrooms, and the personnel to fill training and trainee billets, and the magnitude of training materials (but not the content), become the continued focus of MPT analysis and documentation. Finally, MPT resource requirements documentation contributes to the military integrated logistics support determinations, which begin to take into account the total life-cycle operation and cost of a weapon system or training device.

In sum, MPT front-end analysis and iterative MPT requirements determinations are based on performance data of both the equipment and the personnel. By interfacing with the ISD and technical documentation processes, the analysis establishes a structure of resources within which the equipment operator/maintainer will learn and ultimately perform.

ISD Process

Instructional Systems Development uses initial performance data to determine the trainee population, its training needs, and the constraints that will affect media, mode, training environment, and content decisions. Thus, although the use of the data differs, the initial data base is the same as that required by MPT analysis and technical documentation. The systems model in Figure 1 reorganizes the familiar ISD tasks slightly, and it suggests a product to facilitate an important interface of ISD with MPT analysis: a Training Device Opportunity Report.

It has long been suggested that training experts should play a role in the definition of training devices. Typically, MPT analysts had completed their requirement to determine the training devices before the ISD process had progressed sufficiently to provide substantive input. Even if the ISD process had begun, the mechanism for influencing training device determination was absent or inefficient. This model suggests that after initial data gathering, performance tasks can be identified, and the instructional expert has sufficient data to indicate training device opportunities and constraints that corroborate, modify, or enhance MPT decisions. A subsequent Training Device Opportunity Report further refines and integrates the two analyses contributing to the specification of training devices (e.g., simulator, stimulator, or part-task trainer). This is important because MPT data are based on man's interface with the equipment and ISD data are based on tasks accomplished with or through the equipment. In addition, early consideration of the training device software characteristics (to prompt or vary responses as skills build, to record data, to evaluate, to provide feedback, etc.) can begin, and the instructor console requirements can be better designed.

The Training Device Opportunities Report should perform the following functions for all of the performance tasks:

1. Identify appropriate and inappropriate media which can handle the performance task in a training setting. At this time the analysis should not specify a particular medium; rather the function of this report is to identify plausible media or device options.

2. Relate key task performance characteristics, both conceptual and psychomotor to the training device requirements (e.g., the tactical equipment components or sub-systems, displays, automatic test equipment or built-in test equipment).

3. Identify initial, upgrade and end--point learning support characteristics which the training device needs to prompt, measure and/or report. Initial characteristics might be heavy trainee performance prompts; upgrade characteristics might be "available helps" or less frequent prompts; and end-point characteristics would include only those prompts available on the tactical equipment.

Another interface between ISD and MPT analyses relates the curriculum to the training device. In the past, MPT analysts have made training device recommendations and determinations without the benefit of the identified tasks and educational objectives for trainees, a function of the ISD process. The model suggests that the ISD training analyst can provide the educational justification for a particular kind of training device based on skills, knowledges, and abilities specified at a top-level requirement rather than in terms of more specific statements developed later in the ISD process. This allows for critical MPT and ISD analysis of the impact on the curriculum of the training device to be conducted before instructional strategies and trainee evaluation mechanisms are finalized. Thus, training needs of the population who will use the training device can be accommodated through this MPT/ISD interface.

ISD analysis must also be coordinated with technical documentation development, especially

when technical documentation presents operational and maintenance procedures or equipment-specific displays. Through the ISD process, trainee objectives for acquiring the knowledge and cognitive/motor skills to use the procedures and displays, and the evaluation standards and criteria for acceptable levels of trainee performance are developed. Interface between ISD analysts and documentation writers should be initiated during the system design phase and maintained through the system development and test and evaluation phases as the equipment and documentation evolve. When either technical documentation developers or ISD analysts avoid interaction until system development and implementation decisions have been locked in, inefficiencies and conflicts that waste time and money result.

Technical Documentation

The MPT and ISD process can be developed with great precision, but the element that makes them effective and accurate, and which is often developed in an engineering vacuum, is technical documentation. The question can be posed, "What good does it do to have the right person in the right place at the right time (as a result of a precise MPT analysis), with the appropriate training (as a result of a precise ISD analysis), if he opens his reference document to repair a circuit only to find that it is unreadable and addresses component-level replacement when he has been trained only to the card-replacement level?" Clearly, a key element in the design of personnel and training subsystems is appropriate documentation: It is the glue that holds the MPT and ISD processes together by connecting training to successful operation of the system/equipment. If the documentation is not appropriate to the mission activities (operation, maintenance, installation, transportation) and to the personnel who perform these activities, then it misses its mark. Therefore, it is important that MPT, ISD, and technical documentation initiatives be accomplished in consonance so that the right people with the right training use the right procedures and use them effectively to accomplish their jobs.

With the rapid advancement of military technology, the traditional tactical equipment operation and maintenance personnel are quickly becoming equipment and system "managers" vice "technicians." Recent equipment design has generally focused on having the hardware perform automatically as many operational and maintenance functions as possible, thus relegating the operator and maintainer to the job of "monitoring" to ensure that the equipment or system functions properly. These "higher level equipment management" tasks require much less technically detailed documentation for field use, saving the engineering detail for depot or intermediate-level maintenance. Thus the Equipment Maintenance Concept, Operation Concept, and specifics of Equipment/System Design Parameters require careful analysis in consonance with the initiation of the ISD and MPT Weapon System Analysis processes to ensure the appropriate (1) type of documentation (maintenance, operational, transportation, installation); (2) level of documentation (at the reading grade level appropriate for the anticipated operator/maintainer personnel as well as applicable to the intended task level requirements such as card replacement, component replacement, etc.); and (3) quality of documentation (to withstand rigors of platform use with exposure to continuous use, intermediate maintenance in a controlled environment, or a depot environment). Design of the resulting equipment technical manuals is critical to the design of the curriculum because the manuals form the reference base for most learning activities. Therefore, there is a direct interface with the ISD process. The finished training materials must reflect the equipment data and displays in the final job performance aids, and this relationship must continue throughout the equipment's life cycle via a controlled update process.

CONCLUSIONS

The concept that front-end weapon system design is impacted by MPT analysis and technical documentation data, with subsequent detailed development of resource requirements, training content, and technical documentation, drives this Training Systems Integration Model and establishes its relevance as a structure for integrating training systems development roles. Performance data and operational and maintenance parameters, developed and documented by engineers, are used by manpower analysts to establish resource requirements, and focus the ISD analysts in the training content development process.

The proposed model does not attempt to add still another set of military requirements to those that already exist. Rather, it attempts to suggest where work may occur more efficiently within current requirements to yield decisions that are based on the right input, provided in a timely and efficient manner. Several suggestions emerge from this paper:

- Duplication of the initial data gathering efforts can be avoided with performance data (including weapon system parameters and personnel functions and workloads) generated as one effort.

- Substantive and timely ISD input can contribute to the determination of training devices.

- Coordination of technical documentation requirements with MPT and ISD training requirements will produce training materials that meet the broad range of military trainee needs in a manner that enhances their ability to operate and maintain highly complex tactical equipment.

Additionally, once the proposed model has been implemented and tested, the indicated interfaces could be available via computer, thus making the data manipulation easier and enhancing the coordination among the analyses.

The model we propose is a dynamic mechanism for identifying points of interface; all of the

processes--MPT, ISD, and technical documentation--occur at different times in the Weapon System Acquisition Process and have different emphases. By presenting the Military Training System in its totality, however, the model shows how all involved in personnel and training system development, implementation, and evaluation can interact in a timely and cost-efficient manner. This, in turn, will provide a way for military personnel to operate and maintain tactical equipment with the proper training and for that training to be grounded in those performances established through appropriate analysis, development, and documentation techniques.

REFERENCES

Chief of Naval Operations, OPNAV Instruction 5311.7, Determining Manpower, Personnel and Training (MPT) Requirements for Navy Acquisitions, October, 1985.

Office of the Deputy Chief of Staff for Personnel, AR 602-2, Manpower and Personnel Integration (MANPRINT) in the Materiel Acquisition Process, July, 1986.

Courtice, Andrew J., "The Application of Front-End Analysis to the Acquisition of Air Force Training Systems," Proceedings of the Seventh Annual Industry/Interservice Training Equipment Conference, Orlando, Florida, November, 1985.

Andrews, Dee H., "Ensuring that Front-End Analysis Data Is Utilized During the Design (Engineering) Phase," Proceedings of the Seventh Annual Industry/Interservice Training Equipment Conference, Orlando, Florida, November, 1985.

Department of Defense, Military Standard (MIL-STD-001379C (Navy)). Military Training Programs, December, 1985.

Department of Defense, SECNAVINST 5219.2, Technical Manual Program Management; Policies and Responsibilities for, May, 1983.

Department of the Navy, Strategic Systems Project Office, SSP Instruction 5600.21, Coordinated Development of Tactical Equipments, Technical Manuals, and Training Materials to Support SSBN Personnel Training; Policy for, October, 1983.

ABOUT THE AUTHORS

Dr. Robert S. Bloom, a Senior Training Specialist at GP Taurio, Inc., performs Navy HARDMAN analyses for submarine-specific weapon systems and also provides consultation in monitoring and reviewing Navy training program development. Dr. Renata V. Schmidt, a Senior Training Analyst at General Physics Corporation, coordinates workshops on ISD and Instructor Training. Mr. William P. Hardy, Division Director of the Personnel and Training Systems Division at GP Taurio, Inc., provides technical training documentation expertise to the Naval Sea Systems Command, Space and Naval Warfare Systems Command, and Chief of Navy Technical Training (CNTECHTRA). General Physics and its subsidiary, GP Taurio, are major training systems development and support contractors in the military, nuclear utility, automotive, and commercial training arenas throughout the United States and Canada.

The authors wish to acknowledge the invaluable assistance of Priscilla Goeller whose insights and suggestions helped us to structure our thinking and the presentation of our systems approach. We thank many of the staff of the Personnel and Training Systems Division of GP Taurio for their thoughts and reactions.

INTERDISCIPLINARY SYSTEMS DEFINITION MODEL

Gary J. Ketchie, Ph.D.
Kristen L. Thompson

Honeywell Inc.
Training and Control Systems Division
2855 Anthony Lane South
St. Anthony, Minnesota 55418

ABSTRACT

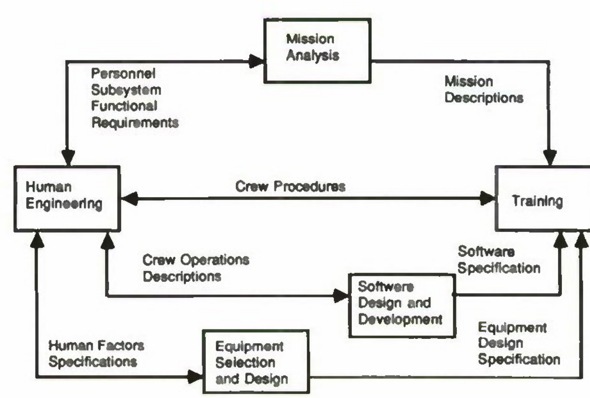
This paper will present information on an Interdisciplinary Systems Definition Model (ISDM) for training design and developments which is implemented during the military acquisition process, and which utilizes a diverse range of technical skills and disciplines. The central theme of the model emphasizes the need for individual technical disciplines to coordinate not only products but processes which may affect an adjacent discipline's methodology. The focus of the model is the definition and development of those aspects to be trained which address the functional and operational aspects of the system. Functional aspects in this context deal with the skills required to place the system into a state of functioning, or simply, the man-machine-interface. Operational aspects refer to activities performed by the operator(s) in response to the changing tactical environment, including coordination and communication with the supported echelon of deployment. In addition, this paper provides information on the systems engineering approach used to define doctrinal deployment and tactical applications of a system with no type classified predecessor or similar system in the field. The model will show how the disciplines of Mission Analysis, Human Factors Engineering, and Training have been brought together to define user applications. In this paper, these factors are considered in the context of the Human Factors, Manpower, Personnel, and Training (HMPT) model which preceeded the current MANPRINT model. This paper will describe how the variables of the battlefield environment, threat, and taskings affect the hardware, software, soldiers, and procedures which determine the overall contribution of the system to force effectiveness. As an example, this paper will show how the model has been applied to the Joint Surveillance Target Attack Radar System (Joint STARS), an evolving system in the Military Acquisition Process. By utilizing the skills of mission analyst, human factors engineer, and training developer, concerns related to work station layout, workload, crew size, sensor performance, and training developments have been addressed during the validation and full scale engineering development stages of the acquisition cycle for Joint STARS. Finally, this paper will show examples of how the inter-disciplinary approach was applied to system and personnel issues which affected software design, operational concepts, and training.

INTRODUCTION

The effectiveness and readiness of a weapon system depends, to a large degree, upon the system operator, crew, and maintainers. Yet, frequently, little attention has been given to human performance capabilities and to Human Factors, Manpower, Personnel, and Training (HMPT) issues during the development phase of new systems.

Because of system effectiveness being dependent on operator, crew, and maintainers, there has been an increased awareness of HMPT concerns within the DoD which is reflected in changes to system acquisition regulations and policies. A greater emphasis is now placed upon the incorporation of HMPT considerations in the planning stage of new systems, as well as during their development, evaluation, and fielding. The changes in DoD regulations and policies focus particular attention on the ability of system design to meet the capabilities of the people who will use the system, and on the availability of adequate numbers of people with the right skills to operate and maintain the system. Further attention is focused on provisions for safe and effective system operation and maintenance.

The Interdisciplinary Systems Definition Model (ISDM) diagram represents the approach used to address HMPT concerns. The approach focuses primarily on Human Factors, Mission Analysis, Training, and System Design.



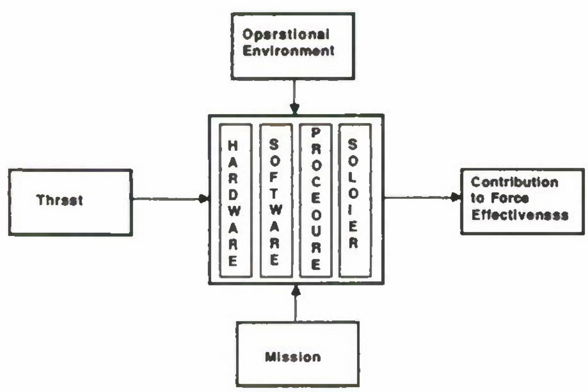
The selection among alternative design concepts involves consideration of human capabilities for information processing and decision making when dealing with system throughput of target information. Performance is considered both for the human as an individual and for the human as a member of an interacting team. The design and evaluation process entails allocating functions between the operator and machine in accordance with human strengths and weaknesses, and providing the operator with job and decision aids to optimize the man-machine interface.

The products of these activities provide the basis for the preparation of design requirements and training needs. Included in the design are

such elements as workspace layout, crew station configuration, and crew composition. The design of the man-machine interface takes into account procedures for processing, manipulating, and transmitting information in terms of human requirements and capabilities. Training requirements are extrapolated so that recommendations can be made for training equipment, support personnel, and facilities.

DEFINITIONAL MODEL DESCRIPTION

In the context of structuring the ISDM, a systems operational concept Definitional Model was defined and is diagrammed below. The Definitional Model allowed for the specific interactions of the ISDM disciplines while permitting interface with the variables that affect the communications and responses between the Hardware, Software, Procedures, and Soldiers.



Procedures/Soldiers

Well defined procedures are essential to ensure adequate training of operators and to ensure the operational adequacy of the hardware and software to meet the user needs. Definition of the procedures are categorized as Functional and Operational and are presented below.

Functional Procedures. Functional procedures are those tasks performed to place the system in operation. These tasks may require interaction with other operators. However, emphasis of these tasks is on the interaction of a single operator with the hardware and software.

Operational Procedures. Operational procedures are those tasks involving more than one operator and/or tasks involving an operator(s) plus a user for accomplishment. Operational procedures generally involve tasks associated with communication and coordination with the user elements in reference to the information provided by the system.

Procedures designate how the soldier is to:

--Convert user taskings into operator functions.
--Filter non-essential information.
--Interface with other battlefield systems.

Operational Environment

The Operational Environment offers unwanted surprises to the operational system. Identification of these surprises and the effects they have on system performance will determine the tasks for which operators need to be trained in order to mitigate the effects on system performance. The Operational Environment considerations include:

Non-linear Battlefield. In modern battle, the US Army will face an enemy who expects to sustain rapid movement during the offense and who will probably use every weapon available. Opposing forces will rarely fight along orderly, distinct lines. Massive troop concentrations or immensely destructive fires will make some penetrations by both combatants nearly inevitable. This means that linear warfare will most often be a temporary condition at best and that distinctions between rear and forward areas will be blurred. Air and ground maneuver forces; conventional, nuclear, and chemical fires; unconventional warfare; active reconnaissance, surveillance, and target-acquisition efforts; and electronic warfare will be directed against the forward and rear areas of both combatants.

Weather. Weather affects equipment and terrain, but the greatest impact is on the soldiers. Perhaps the most important effect of weather is on the soldier's ability to function effectively in battle. Inclement weather generally favors an attacker because defending troops will be less alert.

Airspace Management. Airspace coordination maximizes joint force effectiveness in the battle without hindering the combat power. Friendly aircraft must be able to enter, to depart, and to move within the area of operations free of undue restrictions, while artillery fires in support of ground force continue uninterrupted. The tempo and complexity of modern combat rule out a system that requires complicated or time-consuming coordination. Also, the likelihood of poor or enemy-jammed communications dictates maximum reliance on procedural arrangement. To be simple and flexible, our airspace coordination system operates under a concept of management by exception.

Line-Of-Sight. The application of the Definitional Model requires the identification of visibility criteria among three variables; 1) aerial platform, which is defined as altitude, stand-off distance, and position, determined by time; 2) intervening terrain, which determines masked areas and graying angle for target detection, and 3) shelter location, to maintain maximum line-of-sight between the airborne and ground based data links.

Threat

As defined and utilized within the Definitional Model, the Threat is the Warsaw Pact Forces in general and the forces opposing the United States contingency of the U.S. Corps along the Fulda Gap avenue of approach in particular.

Development of the Army only threat scenario proceeded within the guidelines described by Soviet Army Operations IAG-13-U-78. This document describes the basic flow of maneuver and air deployment patterns and provides an ending location for maneuver units at the regimental level. The Red Force organization depicted is representative of a 1986 time frame. Based on accepted Red Force doctrine, extensive terrain analysis, and the documentation guidelines, specific movement routes,

forward assembly areas, velocities, and formations were defined for each maneuver battalion in the threat area.

Mission

Effective taskings help ensure that the right information is collected to support mission accomplishment while using the least amount of critical resources. Tasking controls the information flow through a system by specifying the information needs in terms of level of command, location, and time.

Level of Command. The deployment of a system to a specific echelon will, by the nature of the threat encountered by that echelon, determine mission types and taskings encountered. Associated with the mission requirements of the level of command are the area of influence and the area of interest.

The area of influence is the assigned area of operations wherein a commander is capable of acquiring and fighting enemy units with assets organic to, or in support of, their command. It is a geographical area, the size of which depends upon the factors of METT-T (Mission, Enemy, Terrain, Troops Available and Time). It is assigned by higher headquarters and designated by boundaries and a forward terminating line.

The area of interest extends beyond the area of influence. It includes territory which contains enemy forces capable of affecting future operations. The area of interest is usually within the next higher headquarter's and a portion of adjacent units' areas of influence. The area of interest contains units not yet closed in battle, but within striking distance of an echelons forces.

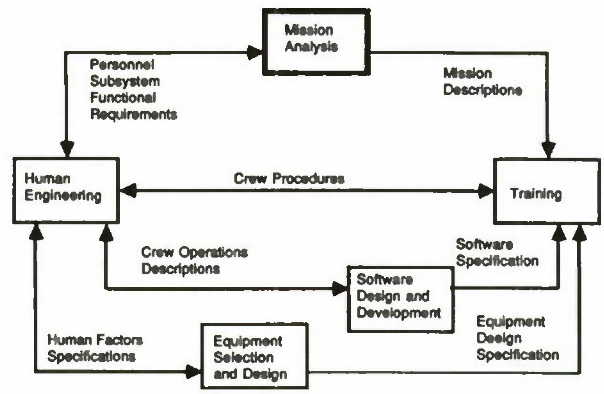
Contribution To Force Effectiveness

The ability of a single system to influence or contribute to the success of an operation must be considered in conjunction with and in the context of its supporting system and the system which it in turn supports. To quantify the contributions of a specific systems intra-actions, the intra-actions must be levied against the parallel, queued, and queuing systems, with which that specific system interacts in the larger operational environment. The range of specific system contribution to force effectiveness is therefore subject to the particular question being asked and the paradigm being addressed. Due to the variety of circumstances and the situational nature of a system's placement, to arrive at "contribution to force effectiveness" the implementation of the operational concept design was exercised only at the intra-action system level. However, this is not to infer that by restructuring the operational concepts during intra-actions outputs that the results could not be coordinated to affect results of play on a larger scale.

ISDM JOINT STARS APPLICATION

ISDM Mission Analysis

The mission analysis portion of the ISDM supported the definition of the Operational Environment and Threat in the system's operational concept design. The purpose of the mission analysis effort was to provide a movement scenario which depicts the activity of a Red Force army conducting a supporting attack as part of a Red Force Front assault on the Federal Republic of Germany.



Development of the movement scenario proceeded within the guidelines described by Soviet Army Operations IAG-13-U-78. This document describes the basic flow of maneuver and air deployment patterns, and provides an ending location for maneuver units at the regimental level. In the development process of the movement scenario, movement resolution is increased by depicting the regimental movement described at the battalion level, thus providing greater detail within the scenario. The overall scenario involves a massive Red Force build-up to and conduct of a breakthrough attack. Red Forces depicted include two divisions and an independent tank regiment in the first-echelon, and two second-echelon divisions. The army that is depicted is a first echelon army in the Red Force attack.

The Red Force organization depicted therein is representative of a 1986 time frame. Based on accepted Red Force doctrine, extensive terrain analysis, and supporting documentation guidelines, specific movement routes, forward assembly areas, velocities, and formations were defined for each maneuver battalion in the scenario. All Red Force units were nationalized and their designations are described in this document as such. Blue Forces were not depicted in the scenario.

An in-depth terrain analysis allowed us to select the most realistic and efficient movement routes. Since Red Forces mass their units in specific avenues of approach, this analysis provided the network from which to control the ebb and flow of traffic.

The documents were researched and analyzed which allowed the extraction of ending locations for regimental-size units at approximately four hour increments based on critical incidents. This information was then compiled into event-based timelines at the regimental level. The event-based timelines were transformed into movement timelines at the battalion level with the use of appropriate terrain maps. This allowed us to select the best suited road network on which to deploy the troops forward. The movement timelines were organized in layers by combat division, which reflected the type of activity: maneuver, artillery, or logistics.

The maneuver overlay of the scenario includes 14 hours of pre-attack build-up. The construction of this build-up was based upon possible Red Force deployment patterns and Red Force strategies depicted in available literature. The main areas researched were deception, surprise, and deployment. The combat troop build-up utilized an FTX wargame area in East Germany from which a deception plan could be established. The build-up began with an eight battalion two-sided wargame already in progress. As the wargame took place, Red Force combat troops from the rear were deployed forward in such a way as to imitate combat support. In total, seven regiments were deployed forward from the rear area. The troops were deployed using major autobahns and existing railroad routes. An attempt was made to show the first-echelon battalions who initially conduct the attack already moving when they reach and deploy from their assigned initial positions described by the documentation.

Deployment of regimental Artillery Groups (RAGs), Division Artillery Groups (DAGs), and Army Artillery Groups (AAGs) supporting the Red Force attack was defined from an analysis of the maneuver posture depicted in the scenario and from Red Force doctrine. Thus, the movements of cannon artillery units between alternate firing positions were defined based on their requirements to support the Red Force attack. Multiple Rocket Launcher (MRL) battery movements were described to reflect their anticipated "run and gun" tactics.

Because the researched documentation does not describe the movements of supply units, an analysis of re-supply requirements was also undertaken. These requirements were then translated into supply unit movements following accepted Red Force logistics doctrine. These movements were depicted in terms of specific arrival and departure times, speeds, and formations of units, traveling along specific movement routes between supply points. Both delivery and return-trip activity were depicted. The lowest level of supply activity depicted was regiment transporting to battalion.

Several additional features have been incorporated into the movement scenario. These include:

- Rail Traffic - This feature was implemented as a means of deploying the first-echelon combat troops and artillery forward from the rear area during the pre-attack period.

- River Crossings - Several river crossings are depicted in the scenario, including the build-up to and the conduct of the actual crossing.

- Airmobile Landings - Two battalion-size airmobile landings are depicted in the scenario.

- Formations - Several new formations have been added to the movement scenario including:

  - Rail Roads
  - Advanced Guard Administrative Columns
  - March-to-Contact Administrative Columns
  - Regimental Headquarters
  - Main Body Administrative Columns
  - Rear Guard Administrative Columns
  - River Crossings

- Semi-fixed Installation (SFI) Signatures - The purpose of the SFI modeling effort within the movement scenario was to represent the movement patterns of lucrative milling targets for both artillery and acquisition functions. The following signatures were depicted in the scenario:

  - Forward Line of Troops (FLOT)
  - Assembly Areas
    - --Battalion
    - --Regimental
    - --Division
  - Command Posts
  - Supply Points
  - River Crossings
  - Special Operations

The scenario depicted the detailed movement of all significant MTI-detectable maneuver, supply, and field artillery units participating in a Red Force Army breakthrough attack during a 66-hour period. It should also be noted that, during the plotting of all maneuver, supply, and artillery movements, care was taken to time-phase unit movements with respect to one another. Thus, the movement scenario sought to realistically depict the ebb and flow of traffic and the use of routes and avenues.

The completion of the operational environment description and threat depiction by the Mission Analysis discipline produced a movement scenario which could then be coded by software personnel. The movement scenario was significant because it was the basis by which simulated MTI imagery could serve as the driver for the Joint STARS Ground Station Simulator (GSS) to present typical threat density arrays to a trained subject population for throughput studies, operator evaluations and system analysis.

The GSS testbed facility was developed to verify Joint STARS deployment concepts and to define operator functions and tasks. The Ground Station Simulator provided hardware and software capability to assist in Human Factors Analysis and the training of Joint STARS ground station individuals and crews in both functional and operational tasks. The GSS had the capability of simulating those functions of the Joint STARS Ground Station Module (GSM) which were necessary for training and analysis of the GSM crews. The bases for all Joint STARS Ground Station functions were the Critical Design Plan (CDP) and the Joint STARS B-5 functional software specification. The GSS incorporated both full and/or part task training features as necessary to train operators in individual, team, and superteam tasks. The GSS also had a subset of the communications linked to Joint STARS users. This subset consisted of those links determined by Honeywell and the Program Office to be necessary for training operators in the defined areas.

The GSS computer based facility consisted of 10 Joint STARS GSM operator student stations, three Joint STARS user workstations, and the computer processor and peripherals necessary to simulate the functions of a Joint STARS ground station and its communications links to users. Five simulated S-280 shelters housed the ten student stations, with two student stations per shelter.

The layout of the GSS allowed it to be operated in any of several different configurations,

including full operating capabilities and degraded modes of operation.

A GSS shelter simulation consisted of two student stations, two digitizer boards, a serial printer, field phones, internal intercom capabilities, and equipment rack mockups. The ten student stations were placed in five shelters of approximately the same size as the Joint STARS GSM to be fielded. The internal layout of the GSS shelter possessed a high degree of physical fidelity with the layout of the Joint STARS GSM.

The combination of the movement scenario and the Ground Station Simulator testbed provided the capability to initiate the analysis of Joint STARS concepts on a total systems level, and of the functional and operational procedures required to accomplish mission objectives.

ISDM Human Factors Engineering

Once the movement scenario had been combined with the GSS, efforts could continue in the areas of further defining the system concept and identifying operator tasks. The lead discipline in this analysis was the Human Factors domain; however, the analysis was structured to utilize the maximum potential of the Human Factors and Mission Analysis integration. During the process of defining the effort, an audit trail was produced for the identification of decisions and tradeoffs between decision elements. The areas addressed by this effort were functional analysis, procedural analysis and effectiveness analysis.

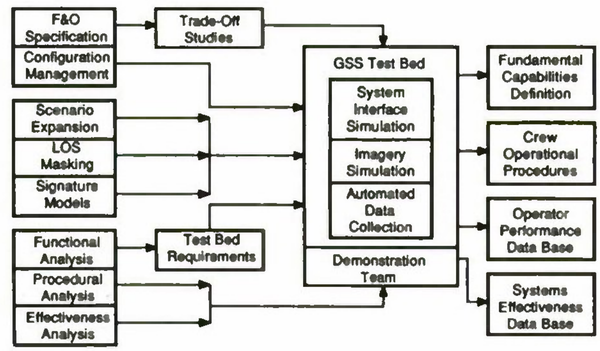
Functional Analysis - All major systems concepts were developed around some stated mission. The proposed mission was analyzed in terms of clarifying its purpose and objectives. These became the underlying basis for all succeeding decisions regarding both the projected hardware and the facility and personnel requirements for the system. Once the general mission purpose and objectives were identified, reasonably detailed operating requirements were defined to clarify the demands to be made on the elements of the system. These requirements were used to define functions that had to be performed by physical elements, such as hardware, facilities, or software, and/or by operators, technicians, maintainers, or managers.

Procedural Analysis - Once baseline functions were defined, various procedural approaches toward functional accomplishment were examined. Objective evaluation criteria were established against which to compare alternative procedural accomplishment methods, modes, or techniques. An important aspect of this procedural analysis was the decision whether certain functions would be performed more efficiently or cost effectively by humans, or by equipment (machines or software).

Effectiveness Analysis - The effectiveness analysis provided the basis for adding appropriate human factors information and/or recommendations to the hardware and software specifications. This analysis focused on developing and quantifying preliminary descriptions of what humans do in the system, how they do it, and what the critical input and output characteristics are between human, machine, and operating environment. The descriptions on which analyses were run included:

- The location of the tasked activity
- The type and amount of information input and output
- Time and accuracy requirements
- The potential failure modes and consequences (including effects on operator performance and potential hazards)
- Operator skill requirements

The interactions of these analyses and the products provided by them established a data base from which alternative concepts and functional or operational procedures could be assessed.



ISDM Training

The information and materials generated during the Mission Analysis and Human Factors Engineering efforts were then used to provide for training a realistic scenario and a ground station simulator that had physical and functional fidelity with the anticipated Ground Station Module. By making use of the Events Detection and Sorting Routines software and interdisciplinary information exchanges throughout the system development process, training scenarios that allowed for accurate evaluation of operator performance against known ground truth were developed. The implemented training approach was a seven-staged process leading to the development and conduct of a total of 110 lesson plans for the Joint STARS training package. The lesson plans called for 360 Instructor Contact Hours, of which over 75% were designed and used for hands-on performance.

In developing the training package, well defined procedures were essential to ensure that operators would be adequately trained to use the system efficiently and effectively. In the ISDM, defining these procedures required the application of the functional and operational procedural areas of the systems operational concept. To develop the training package, then, a systems engineering approach that involved seven stages was used to look at both the functional and operational aspects of the proposed system. Those stages were to:

1) Review Materials
2) Develop Total Task List
3) Develop Critical Task List
4) Develop Course Outline
5) Develop Lesson Plans
6) Provide Lesson Analyses
7) Develop and Maintain Administrative Documentation

Review Materials - The review process was begun the moment the original specification materialized for the system under development. Along with the mission analysts and human factors engineers, training personnel worked with the system specification to gather information about the system objectives in each area of expertise. This

process led to many discussions and to the development of throughput, or trade-off, studies to further define potential functional and operational procedures in each of the interdisciplinary areas. Many of the results of these studies were directly folded into the training development process.

The Joint STARS program had additional information available for review since Joint STARS evolved from the PAVEMOVER and I2 SOTAS programs. Preliminary use task lists for the SOTAS Ground Station operators yielded information concerning the functional tasks required to operate the SOTAS Ground Station equipment in the operational environment. Although the displays, controls, hardware, and software characteristics of the Joint STARS were substantially different from those of the I2 SOTAS, many of the operational tasks (those activities needed to perform missions and to carry out taskings) were identical.

Develop Total Task List - The result of the review process was an inclusive list of tasks required of system operators. This list encompassed tasks that were both functional and operational in nature; however, the list did not look at the criticality of the tasks. The task list did not include any procedural narrative at this stage; rather, tasks were defined at such a level that minimal narrative was needed to describe actions associated with a task. As a structural and developmental vehicle, these tasks were formatted into a sequenced training topic list for each operator of the Joint STARS.

Develop Critical Task List - The total task list was then subjected to determinations of each task's criticality toward mission success. The determination of task criticality was based on a modified Delphi technique using the Training and Doctrine Command (TRADOC) Four-Factor Model. People most knowledgeable about the subject matter under evaluation were identified to evaluate each area of the total task list using the Four-Factor Model. Discrepancies among these experts were then resolved through discussions, resulting in the compilation of a list of the tasks considered most critical to efficient and effective system operation. These critical tasks to be trained were further categorized into functional and operational tasks and sequenced for potential course conduct.

The critical task list produced by these experts also provided the opportunity to evaluate the most appropriate media with which to train the critical tasks. This media selection process took into account the tasks, the identified-operator skills (skills defined by Military Occupational Specialty) and potential-task familiarity (prerequisite skills), the effectiveness of various training methods given the defined tasks, and the training media available at the defined training site. With these considerations, training media were identified for the Joint STARS Operator's Course.

Develop Course Outline - The information necessary to define a course sequence for system operators resulted from the sequenced list of critical tasks to be trained. After the categorization of tasks into functional and operational groups, the course flow established progressed from individual tasks to team tasks to superteam tasks.

Individual tasks evolved around the individual operator learning the functional aspects of the system's hardware and how to manipulate the software efficiently and effectively. The team tasks built on the individual and functional tasks learned by individual operators and combined those skills with the operational aspects of working with another operator inside the same GSS or GSM. The course sequence culminated in training the superteam tasks. The superteam tasks trained the operators in the operational aspects of coordination and communication with the outside user community needed to result in successful mission completion.

Develop Lesson Plans - After course sequencing was defined, the narratives required to support the teaching of critical tasks were developed. For the Joint STARS program, lesson plan development culminated in 110 lesson plans. These lesson plans each had up to three parts: a classroom conference, a self check test with answers, and a hands-on simulator practice script.

The classroom conference represented a clearly stated and measurable task, condition, and standard. Many of the standards were easily attainable as a result of the work performed by the human factors engineers' throughput studies and the threat scenario defined by the mission analysts.

The self check test presented questions on the more important points covered in the conference. An answer sheet was provided so answers could be checked immediately by the student operators. The self checks also were used by instructors to discover which procedures were found by students to be confusing. This information was then folded back into a revision of the course, or documented for later course revisions.

The hands-on simulator practice script reinforced the task covered during the conference. This hands-on time by the student operators allowed for practice with equipment and conditions that would be immediately transferable to the actual Ground Station Module. The hands-on portion of the training course comprised over 75% of the total training time. For the individual and team training portions of the course, the student-to-instructor ratio in the simulator was not more than two students to one instructor. During the superteam training the student-to-instructor ratio increased but was never more than five to one.

Provide Lesson Analyses - The Joint STARS Ground Station Simulator was designed to collect information that allowed instructors to assess how well students were performing functional tasks. For example, keypress data were collected by the system for each student and for each lesson run. These data could then be analyzed to define the keypress patterns used and the number of times specific keys were pressed. Information of this nature provided the opportunity for instructors to detect and change ineffective and inefficient keypress sequences. These data also provided that opportunity to eliminate some of the "superstitious behavior" that can develop when learning on a developmental system.

In addition, student performance in an operational context was readily measurable as a result of the baselines developed during the throughput studies, and the trainers' knowledge of the ground truth and tactical situation resulting from mission analysis and scenario development. These data about the functional and operational system usage were useful not only to identify the

improvements needed for a particular training session but also to identify improvements to be folded into the next revision of the training course.

Develop and Maintain Administrative Documentation - One important aspect of the ISDM is the use of audit trails to document the results of information ascertained by each of the interdisciplinary areas. Communication is critical when using the ISDM so each area of expertise knows what the other areas are working on and how information is being implemented by other areas. Informal communications worked well for the Joint STARS program until a baseline hardware design, software configuration, and training course had been developed. At that point, because changes to the hardware or software could directly affect the development of the training course lesson plans, as could a change in the functional or operational requirements for the training course affect the hardware or software design, a more formalized documentation approach was required. This resulted in the development of Programs of Instruction (POIs) for the training of the main Joint STARS operators. In addition, technical notes were compiled to document specific aspects of hardware design, software implementation, scenario changes due to updated threat information, and the results of continuing human factors studies.

Summary

This paper has described how the disciplines associated with Mission Analysis, Human Factors, and Training have been able to exercise their specific areas of expertise and influence in a definitional model. The paper showed that each of these domains contributed significantly to the overall success of the effort without compromising a supporting area of the investigation. The success of the ISDM application to the Joint STARS program needs to be evaluated against a standard of measure which is greater than the sum of the parts. Because of the high level of communication between disciplines, the definitional and developmental efforts of one discipline were enhanced by the implementation, administration and interdisciplinary interactions. Although the quantity and quality of the communication is difficult to measure, total ISDM products were provided which contributed to the progress of the Joint STARS program. Among the numerous deliverables were: the Functional and Operational Specification, the build and delivery of a Joint STARS simulator; the development of a nine week Joint STARS operator course of instruction, and a trained cadre of military Joint STARS instructor personnel. The ability of each discipline to effectively contribute its expertise to the total effort was enhanced as a result of the channels of communication described within the ISDM.

The implementation of the ISDM at the initiation of the validation phase of the Life Cycle System Management Model provided a mechanism which identified, defined, and described, in quantitative terms, Functional and Operational tasks for the Joint STARS (Army) system. The description of operator tasks and system functions has also served as the foundation for the nine week Joint STARS (Army) Operator Course. This course trains both Target Surveillance Supervisors (TSS) and Search Track Operators (STO) in the functional tasking and operational skills necessary for GSM operation. Because of the systematic procedure for the course development, changes and revisions made to the GSM hardware/software configuration and deployment concepts have been documented and incorporated into the course lesson plans. Currently it is anticipated that the nine week Joint STARS (Army) course will be validated and verified during Instructor and Key Personnel Training and Player Training for DT/OTII evaluation.

Acknowledgements

The authors would like to acknowledge the efforts and contributions made by Ms. Lisa Carlson toward the completion of this paper. The assistance she provided with her editorial expertise was greatly appreciated.

References

Hollingsworth, S.R., Ogden, W.C., and Wingert, J.W., "Functional and Operational Characteristics of the Stand-Off Target Acquisition System (SOTAS): Console Functions for Ground Station Operators (U)," Honeywell Systems and Research Center, Minneapolis, MN.

Daniels, R.W., Collyard, K.B., Ketchie, G.J., Hoffman, J.A., and Thompson, K.L., "Preliminary Use Task List for SOTAS Ground Station Operators - Draft Report," Volume I: Preliminary Use Task Lists for SOTAS Ground Station Operators - Technical Report, Volume II: Appendix B - TSS Individual Ground Station Operator Use Tasks, Volume III: Appendix C - STO Ground Station Use Tasks, Volume IV: RTO Individual Ground Station Operator Use Tasks, Honeywell Systems and Research Center, Minneapolis, MN.

Hollingsworth, S.R., and Wingert, J.W., "Human Factors Design Standards for the Stand-Off Target Acquisition System (SOTAS)," Honeywell Systems and Research Center, Minneapolis, MN.

Bosl, S.B., "Menu Editing User's Manual," Honeywell Systems and Research Center, Minneapolis, MN.

Ketchie, G.J., Collyard, K.B., Gauthier, G.L., Lundblad, J.T., Foss, F.C., Thompson, K.L., Bush, B.K., Oakes, J.M., Bagnall, J.F., Luczka, P.J., Gerber, M.E., Dahms, L.A., and Blackowiak, P.F., "Joint STARS Operator's Course, Technical Note, Volume I - Individual Tasks, Volume II - Team Tasks, Volume III - Super Team Tasks, and Volume IV-Concept Integration Exercises," Honeywell Systems and Research Center, Minneapolis, MN.

Ketchie, G.J., Plocher, T.A., Dolby, M.B., Gerber, M.E., Lynn, B.K., and Oakes, J.M., "Technical Report: SOTAS Enhanced Scenario Report Simulation Update/SOTAS Movement Scenario, Volume I - DRAFT (U)," Honeywell Systems and Research Center, Minneapolis, MN.

About the Authors

Dr. Gary J. Ketchie is a Senior Principal Training Analyst at Honeywell's Advanced Systems Facility, Training and Control Systems Division. Dr. Ketchie has served as the Joint STARS Operator Training Manager since 1982. Prior to joining Honeywell in 1980, he was Branch Chief of New Systems Training at the U.S. Army Intelligence Center and School. Dr. Ketchie has an M.A. in Psychology and an M.A. in Administration, an Education Specialist (Ed.S.) Degree, and a Ph.D. in Education.

Kris Thompson is a Systems Training Analyst for Honeywell's Training and Control Systems Division. She works in the Advanced Systems Facility in Minneapolis, MN, and has been involved in the areas of mission analysis, human factors, and training development and conduct on the Joint STARS program since 1980. Ms. Thompson is presently responsible for the implementation of an Instructor's Console to automate the currently manual aspects of Joint STARS training session preparation and analysis. Ms. Thompson holds a Bachelor's degree in Psychology from Hamline University.

COMPUTER-ASSISTED INSTRUCTIONAL SYSTEMS DEVELOPMENT/LOGISTIC SUPPORT ANALYSIS INTERFACE FOR C-17 AIRCRAFT

H. Barbara Sorensen, PhD
Air Force Human Resources Laboratory
Manpower and Personnel Division
Brooks Air Force Base, Texas 78235-5601

ABSTRACT

The development and delivery of military training on new weapon systems is dependent on the identification of training system requirements early in the weapon system life cycle. An automated interface between Logistic Support Analysis (LSA) data and the Instructional Systems Development (ISD) procedures will provide training developers with a means to assist in identifying training requirements earlier in the weapon system acquisition phase. This paper discusses the design and development of such an interface for the C-17 aircraft being developed by McDonnell Douglas Aircraft Corporation. The interface development includes three objectives: (a) tailoring of an existing computer-aided LSA data system for an emerging weapon system; (b) developing automated ISD worksheets; and (c) demonstration of a prototype interface of the ISD automated worksheets with the aircraft system LSA engineering data. The implications of the ISD/LSA interface are twofold. First, it will aid in the development of training by providing a more efficient method of identifying training requirements earlier in the weapon system acquisition process, and second, it will provide an audit trail for LSA and ISD data being utilized in training requirements development.

INTRODUCTION

An essential requirement for effective military training is the early identification and analysis of training system requirements for emerging weapon systems. Currently, training developers encounter problems with late-to-need logistic support data in addition to non-existent, inadequate, and inaccurate data acquired during the Instructional Systems Development (ISD) process.

INSTRUCTIONAL SYSTEMS DEVELOPMENT (ISD)

Instructional Systems Development (AFM 50-2), as utilized through the Air Force weapon system acquisition process, provides an analytic approach to the decision-making process for planning, developing, and managing instructional programs. The rationale for all training and instructional programs must be documented by developers, managers, and commanders throughout the development process. In order to adequately present this rationale, an in-depth analysis of detailed job and task information must be accomplished. Through the ISD process, initial training requirements are identified from existing job data and analyses from the field, engineering data from the contractor, and judgments on the part of training developers. This process ensures that the training needs for critical tasks will be met.

The outcomes of applying ISD assist training developers in determining what to train, how to conduct training, and how to evaluate what was trained. Sound rationales for these decisions are benefits that result from applying the ISD process. In addition, the training developers can make appropriate decisions on the optimum approach to training applications and technology through the ISD assessment of alternative approaches and solutions. The capability of training developers to make these decisions is being hindered, however, due to the application of current ISD procedures being data-intensive, time consuming, and paper-and-pencil dependent. As a result, the identification and development of training requirements in the early stages of weapon system acquisition is greatly delayed. Other problems that result from the paper-and-pencil application of ISD to training development include continual rewriting of non-standard forms and trainer developed forms, and a lack of documentation to support the training developers' decisions. The solution to this problem is being sought in the development of procedures to automate the ISD process using integrated logistics support and engineering data.

LOGISTIC SUPPORT ANALYSIS

The Logistic Support Analysis (LSA) process, applying scientific and engineering principles to the acquisition cycle, integrates the design and support concepts to comply with the operational needs of the system. Many of the current weapon system acquisitions require that training data be provided through the Logistic Support Analysis Record (LSAR) which is governed by Military Standard 1388-2A. The LSA process is conducted on an iterative basis throughout all phases of a weapon system life cycle to accomplish the support analysis objectives. The intent of this standard is to achieve joint service acceptance of standard requirements, data element definitions, data field lengths, and data entry requirements for the LSAR data. Weapon system information generated by LSA during all phases of the weapon system life cycle is used as an input to follow-on analyses and as an aid in developing logistics products. It should be pointed out, however, that the LSA documentation must be tailored to each specific weapon system in all phases of the system life cycle.

Integration of MPT Data

Early identification of training requirements is dependent on the integration of manpower, personnel, and training (MPT) data in the initial stages of the weapon system life cycle. One of the goals in weapon system acquisition programs is to increase both human and hardware performance. This can be accomplished if programs are initiated early enough for cost-effective front-end analyses (FEA) to be conducted. The integration of logistics, manpower, personnel, and training analyses and data can be realized through FEA.

FEA would enhance the effects of training requirements identification. A source for all this MPT data, if delivered to the training developers in a timely manner, is in the LSAR data. Needed information for the MPT decisions as they relate to both maintenance and aircrew training capabilities can be obtained from 24 of

the LSAR data records and their associated reports. The flow of information needed to feed the MPT utilization requirements for weapon system acquisition programs is shown in Figure 1.

The Manpower, Personnel, and Training Analysis reports assist in the timely identification of the technical tasks that operators and maintainers perform. In addition, it identifies job descriptions, employment doctrine, personnel requirements, the support concept, maintenance and repair systems, and operational manpower requirements. Additional report information includes specified data as skills needed, frequency of task performance, time to perform the tasks, personnel required, location, and a description of the task steps required to complete the performance of the task. (DI-ILLS-80077)

A listing of the minimum requirements of all knowledge and skills required for personnel to effectively operate and maintain a system or subsystem is provided in the Personnel Performance Profiles. These profiles also provide the knowledge and skills to perform a task or function. These profiles can be used to determine training requirements, develop personnel evaluation criteria, standardize training material, develop course objectives for curricular and training material, and minimize duplication of reporting knowledge and skills. (DI-ILSS-8078)

The documentation for the Training Path System identifies the training requirements for all categories of personnel in a training program, thus ensuring the effective development of skills and knowledge necessary to coordinate, direct, and perform operation and maintenance of a system. (DI-ILSS-80079)

Data to evaluate the extent to which equipment having an interface with maintenance meets the human performance requirements and the human engineering design criteria is provided by the Human Engineering Design Approach Document--Maintenance. This document utilizes several records from the LSAR in conjunction with applicable sketches, drawings, and photographs to satisfy the human-equipment interface evaluation. (DI-H-7057)

COMPUTER-AIDED ISD

State-of-the-art technology provides the potential to alleviate some of the problems with the current procedures used by training developers of new weapon systems. A system is needed that will automate the LSA data and allow for tailoring of the data to conform to the requirements for the aircraft. In addition, the system must provide the ability to annotate additions, deletions, and changes on the LSA data being provided by the contractor. Also as ISD is required for educational and training programs, this system must accommodate the entire ISD process. A primary requirement of the CAISD system is to adapt new design information into the ongoing ISD process, to include engineering data and data from system specialists. Figure 2 depicts the flow of LSAR data required to feed the ISD training model.

The Computer-Aided ISD (CAISD) is currently being developed to create an interface between LSA and ISD data to facilitate the ISD process being used by the training developers on the C-17 aircraft. This system includes three components: (a) tailoring of the Computer-Aided LSA (CALSA) system to interface LSA/ISD data for use by the 3306 Air Training Command's (ATC) Test and Evaluation Squadron; (b) the development of automated ISD worksheets that incorporate engineering data/documentation and provide convenient access procedures; and (c) feasibility test, demonstration, and user training on the prototype systems. In addition to designing procedures for developing new technologies into the ISD process, CAISD will support existing state-of-the-art technology by implementing the government-owned Computer-Aided Logistic Support Analysis (CALSA) system.

Computer-Aided Logistic Support Analysis (CALSA)

CALSA is a centralized and automated Logistic Support Analysis Record (LSAR) developed for the government by Dynamics Research Corporation. CALSA has previously been implemented by the U.S. Army and Air Force Government Surveillance and Target Attack Radar System (Joint STARS) program, the U.S. Navy MK 50 Torpedo program, and other LSA defense programs that substantiate its use as a flexible and easy way to use the tailoring system for LSA.

CALSA can be tailored to meet the logistic needs of an particular weapon system. The tailoring is accomplished by a user who manipulates the functions of CALSA. These functions allow the user to accomplish the following: (a) enter and revise data; (b) generate reports; (c) compare different data bases and list the differences; (d) generate models; (e) perform administrative duties, and (f) manage the system. CALSA is an essential component required for the timely and cost effective development of the CAISD. In addition to the specifications required in MIL-STD 1388-2A, CALSA can also serve as an integrated data base for LSA and for other program elements such as ISD.

Tailoring of CALSA

The CALSA data system will be tailored for Air Training Command's (ATC) 3306 Test and Evaluation Squadron (TES), for use on the McDonnell Douglas C-17 aircraft LSA data. One of the missions of the 3306 TES is to determine weapon system aircrew and maintenance training requirements during the early stages of weapon system acquisition. A basic assumption of the 3306 TES is that training, regardless of its setting, should result from an ISD analysis of the weapon system requirements.

Training developers from the 3306 TES determine these training requirements through the ISD process using hard copy LSA data from the C-17 contractor. In order to tailor the ISD model to the objective of identifying training requirements, the squadron has developed a 14-step process for that purpose. Nine of these steps that directly impact on the early identification of training requirements are discribed as follows:

1. Identify system maintenance requirements--all of the duties and tasks that make up a job are identified to include the

Manpower, Personnel, and Training Flow
from Logistic Support Analysis Records

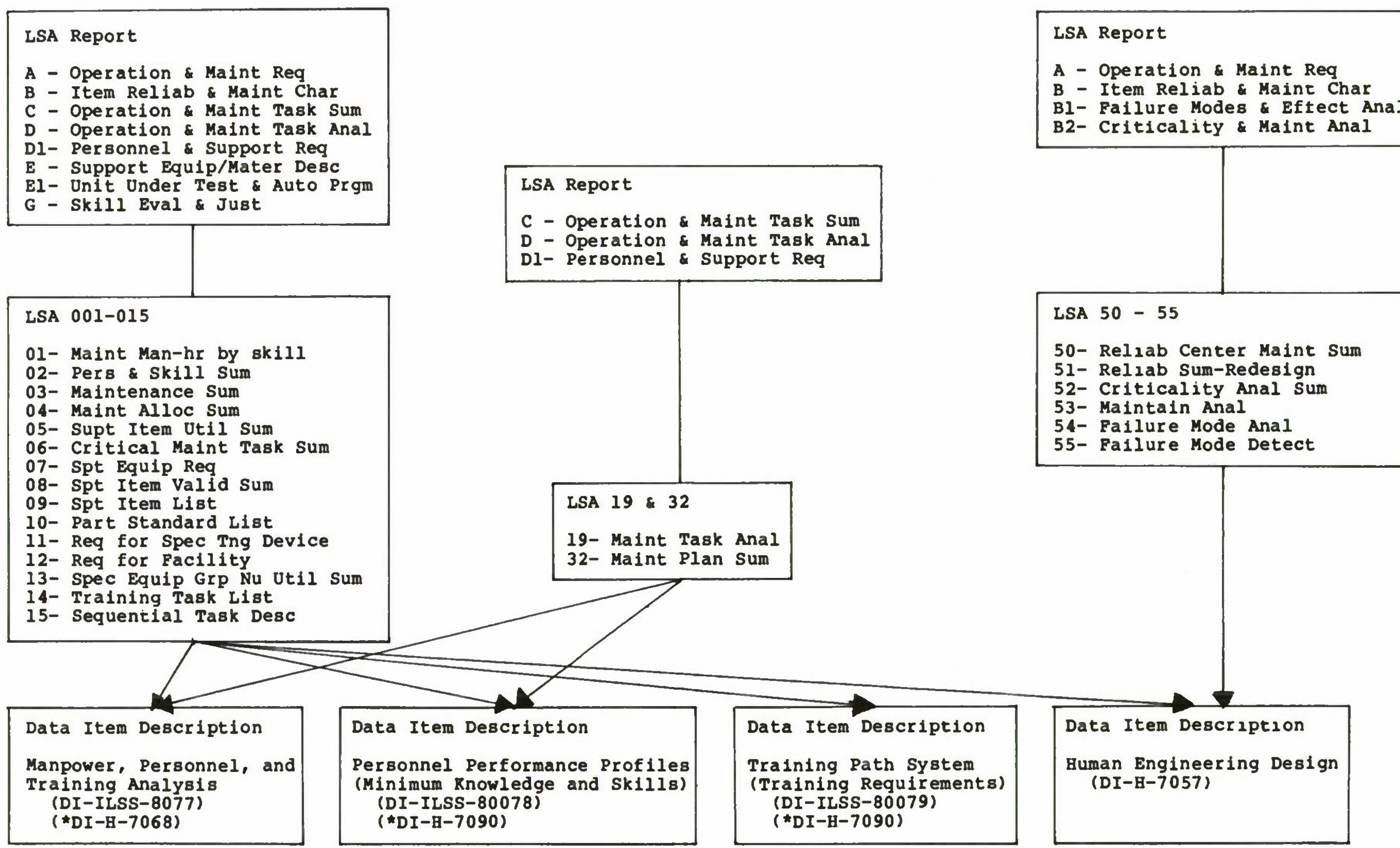


Figure 1.

Specified Weapon System LSAR
Input to ISD Training Model

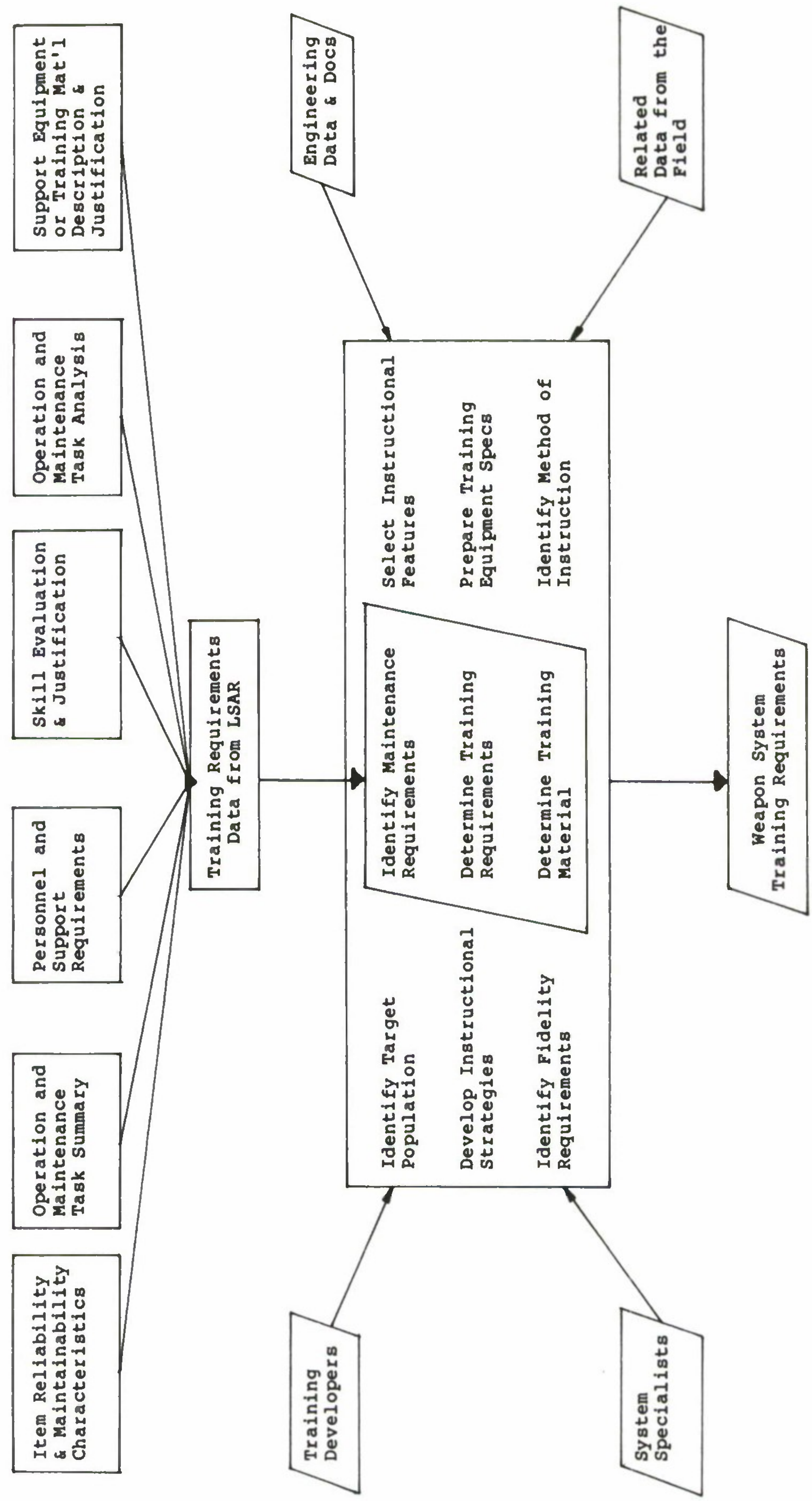


Figure 2.

mission and equipment used. Data gathering for this task list includes identifying all duties, identifying and recording task statements, verifying the task list, and developing group tasks. An end product of this step is the development of job performance requirements;

2. Identify characteristics of the target population--characteristics of the students who are to be trained are determined based on the target population or estimated target population provided by the using command. The target population definition includes the students' entering AFSC and all prior WS experience. Familiarity of skills and knowledges of the target population are obtained from training course/standards, occupational surveys, and subject matter expertise;

3. Determine training requirements--decisions as to what is and is not to be included in training are made based on the difference between results of Steps 1 and 2. Activities of remaining tasks and skills are assessed for potential training requirements. Those activities not eliminated are then matched with appropriate skills and knowledges associated with it;

4. Determine types of technical training material required--determination is made on how the identified skills and knowledges (step 3) can best be acquired by the students. This determination assesses training modes such as hardware, visual aids, printed material, and computer-assisted instruction;

5. Develop instructional strategies--this step focuses on the development of a preliminary overview of the entire training program. Each task for an identified training requirement is analyzed on a separate worksheet that includes the preliminary criterion objectives, a draft of the media description, a brief instructional strategy, and the sequence tasks;

6. Identify fidelity requirements of hardware components--the degree of fidelity of hardware to train specified skills and knowledges is determined by how realistically the hardware must be represented to achieve those training requirements;

7. Select instructional features for hardware media--this step is performed only on sophisticated trainers or where there is complicated student interaction. The four components, steps, or aspects of learning principles are assessed; stimulus, response, feedback, and next activity;

8. Prepare ISD-derived training equipment specification--this is the model for recording the training equipment design written in a military specification format. This model includes the training objectives, training application, simulation characteristics, instructional features, and trainer configuration; and

9. Identify method of instruction--this step includes the selection and identification of the method of instruction for each behavioral requirement based on the media class selected to teach each skill or knowledge. A draft course chart for entire training programs is developed. Estimations of lesson times, block times, and total course length are determined.

Functional specifications for a CAISD process, using the C-17 LSA data to support the ISD decision-making process, will be determined by the training developers at the squadron. Subsequent to that endeavor, automated ISD worksheets, incorporating engineering data/documentation, will be developed. These automated worksheets will provide the user with the capability to globally search and update data, to include the ability to identify the currency of entered data and whether or not it is the most recent available. The user will also be able to integrate LSA data with individual system expert judgments. The feasibility of interfacing the automated ISD worksheets with the C-17 LSA and engineering data will be assessed through a prototype CAISD system. Final functional specifications shall document the approach in the development of the C-17 ISD management information system. These specifications will recommend a design approach to implement the LSA/ISD interface.

DISCUSSION

The development of the CAISD interface with LSA will greatly enhance the performance of training developers in their requirement to identify and document initial training requirements for a new weapon system early in the system life cycle. Although this interface is currently being developed using data for the C-17 aircraft, it is designed for general applicability to other weapon systems that possess LSA.

CAISD provides two main benefits for the development of military training on new weapon systems. First, this interface system will automate and streamline the process of identifying training requirements from LSA using the ISD model. This process will be more efficient in that training developers will be able to access, manipulate, and interact with LSA data through the ISD model in a computer-assisted mode, rather than performing these functions in a lengthy and cumbersom paper-and-pencil mode. Thus, training developers will have a state-of-the-art, efficient technology to assist in identifying training requirements earlier in the life cycle.

Second, CAISD will provide an audit trail of the training requirements identification process. This will allow training developers to accurately document their decisions. In addition, documented, easily accessible data will be available for system reviews.

Early identification of training requirements in the weapon system life cycle directly affects the ability to develop and deliver military training for the maintenance and support of weapon system prior to delivery. The CAISD will assist in the identification of these training requirements.

REFERENCES

AFM 50-2, Instructional Systems Development (ISD), 31 July 1975.
AFR 50-8, Policy and Guidance for Instructional Systems Development (ISD), 6 August 1984.

AFR 50-11, Management of Training Systems, 10 August 1984.
Joint Stars CALSA-Computer-Aided Logistic Support Analysis, Naval Oceans System Center, September 1986.
MIL-STD 1388-2A, DOD Requirements for a Logistic Support Analysis Record, 20 July 1984 and Notice 1, February 1986.
Procedural Handbook, 3306 Test and Evaluation Squadron (TES), January 1985.

ABOUT THE AUTHOR

Dr Sorensen is a research scientist and psychologist at the Air Force Human Resources Laboratory. She has a Masters in Educational Psychology, Measurement, and Statistics, and a Doctorate in Instructional Design and Computer Technology. Her 14 years of education and training systems experiences include the design, development, implementation, and evaluation of those systems throughout her career with the Army, Navy, Air Force, and civilian communities. She began working for the Air Force in 1985 and in addition to other Laboratory projects, currently heads a joint-service project responsible for designing a tri-service weapon system maintenance training interface for instruction through the UBD2000.

SIMULATION FIDELITY: A RATIONAL PROCESS FOR ITS IDENTIFICATION AND IMPLEMENTATION

Robert C. Bruce, Ed.D.
AAI Corporation
Instructional Systems Development
York Road and Industry Lane
Cockeysville, Maryland 21030

ABSTRACT

The degree of fidelity required in simulators to effectively transfer newly acquired skills between the classroom and the work world remains illusive and ill-defined during the front end analysis of system design. Frequently, fidelity specifications are inconsistent between the ultimate users of the system, the acquisition agency, and the contractor charged with the design and production of the final training system. Such a situation is not in the best interest of the student and is likely to produce a device insensitive to the directions provided by sound instructional and engineering analyses. This paper presents a technique for allowing individual training tasks to define specific degrees of simulator fidelity and then objectively tracking the task/fidelity relationship throughout the design, development, and testing phases.

INTRODUCTION

For over 25 years, AAI Corporation has been responsible for the design and production of training devices requiring various degrees of fidelity. During this period, it became apparent that terms such as 100-percent fidelity, full fidelity, etc, were an enigma. In an effort to define the issue of simulator fidelity, or more precisely, "How much fidelity is enough?", AAI developed a model that would allow the training requirements to design the ultimate training systems. In this fashion, if the simulator could totally support the tasks for which it would be held accountable, then the inherent level of fidelity was sufficient.

Defining these training requirements took the combined expertise of personnel with training and systems engineering backgrounds. Using the instructional systems development (ISD) approach, the entire process was conducted in parallel to simultaneously identify tasks to be trained and required levels of task fidelity. Extensive analysis was continuously performed on the collected data to further refine them to a point where valid design decisions could be made.

FRONT END ANALYSIS

To derive the fidelity requirements for each simulator, a comprehensive front end analysis (FEA) was conducted. AAI's research effort initially concentrated on the accurate definition and validation of critical tasks to be trained. These tasks were collated and refined to develop a preliminary task list for each anticipated simulation work station type. A team of training specialists and systems engineers then visited operational sites to observe job incumbents, interview subject matter experts (SME's), collect additional data, and validate the critical tasks to be taught on the training system. The product of this effort was a validated list of training tasks that would have to be supported by the ultimate training system.

After the results of the data research, collection, and validation efforts were analyzed and refined, each training task was classified to a learning category. Each learning category defined the optimum testing environment and instructional delivery media to satisfy the learning activity of each training task. The appropriate medium for each task was selected and documented in a Media Summary. These data were then resorted by subsystem for each job and aggregated to the highest level of fidelity required to support all of the tasks taught on each individual panel. These data were documented in a Panel Summary. The fidelity requirements for each of the equipment panels were combined to define the fidelity requirements for each simulation work station as determined by the associated training tasks.

FIDELITY DEFINITION

Rouse (1982) defined fidelity as "the precision with which the simulator reproduces the appearance and behavior of the real equipment." Hays (1981) proposed a similar definition as "the degree of similarity between the training simulator and the equipment being simulated in terms of its physical and functional characteristics." Massey (1986) has added additional clarification by identifying two basic types of fidelity: physical and nonphysical. AAI has translated the physical and nonphysical into specific requirements of physical and functional fidelity and added a third requirement of task commonality. To illustrate AAI's definition of fidelity requirements, the reader must first understand the relationship between the components of the job that the student will ultimately be required to perform at the completion of training. Each component requires its own degree of fidelity support. This relationship is shown on Figure 1 and is discussed in the following paragraphs.

As observed on Figure 1, the identifier 1.B.55.1 equates to the JOB of On-Line Maintenance Technician, the DUTY of Implementing Preventive Maintenance Procedures, the TASK of Cleaning Unibus Expansion Box, no SUBTASK, and ACTIVITY of Preparing the Unit for Cleaning; therefore, task number 1.B.55.1 will always equate to preparing the UEB for cleaning, regardless of where this activity is documented (e.g., preliminary task lists, learning hierarchy, learning objective, course outline, or tests). The Tasks Listings Report output (Figure 2) shows this information in the following manner.

The fidelity requirements for each activity are summed at the task level. When all tasks within the duty have been defined, the corresponding fidelity requirements are summed to define 100-percent fidelity for that duty category. All duty categories are ultimately defined and the total fidelity requirements are then easily gathered to drive the design of the individual simulation work stations.

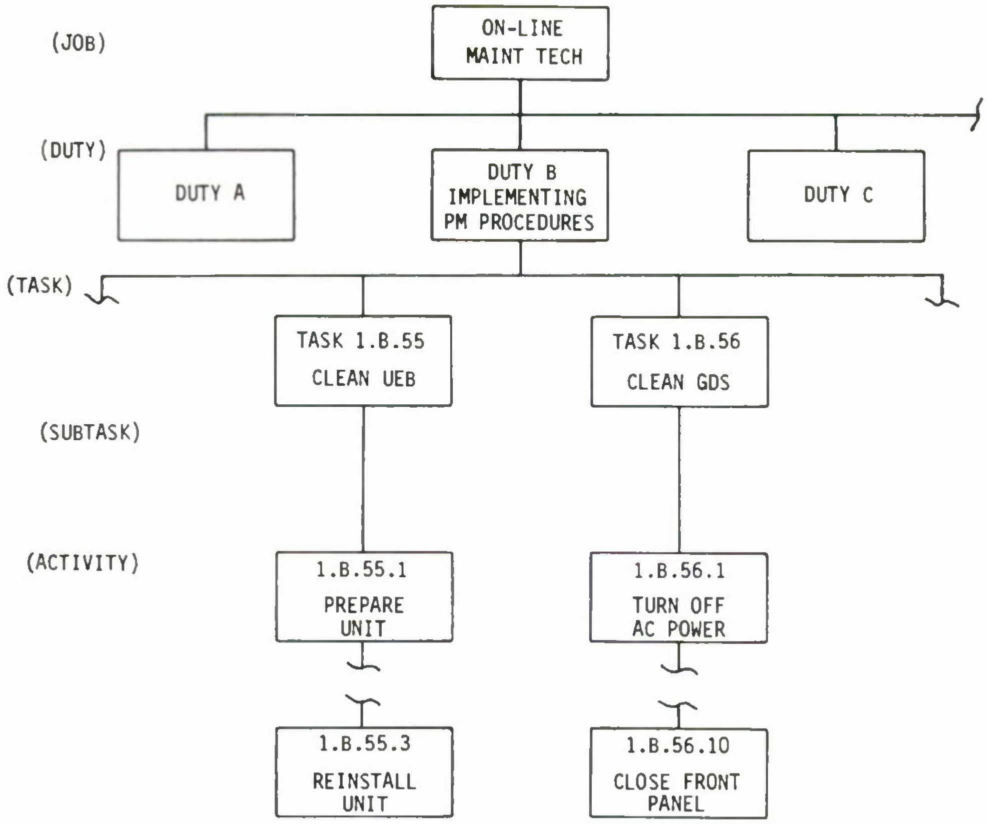


Figure 1. Relationship Among Job Components

The above example has been presented from a top-level perspective without regard to the technical specifications required of the design engineers. It does, however, serve to illustrate the method of identifying fidelity requirements at the lowest level, gathering fidelity issues as the design personnel work up the training components, and ultimately specifying the total simulation work station at the job level.

Each required training task is supported by 100-percent physical and functional fidelity required to ensure that the students gain the skills necessary to do their job in the field. This 100-percent degree of fidelity is translated by the design engineers into specific hardware and software requirements at the lowest activity level. In other words, if a student is required to manipulate a series of events by pressing buttons, changing switch settings, or entering keyboard data, the student work station will provide a simulated operational environment that presents 100 percent of the physical and functional fidelity associated with the real-world cues, responses, and feedback to provide the direct transfer of newly acquired skills into the work world. The following section provides an example of how this is implemented and the fidelity requirements verified throughout the design and development process.

FIDELITY VERIFICATION (FV) MODEL

AAI's fidelity verification approach was modeled after a method for evaluating training device effectiveness reported by the U.S. Army Research Institute for the Behavioral and Social Sciences (Tufano and Evans, 1982). The reported TRAINVICE-A (TV-A) model was modified to facilitate a predesign approach vice the original intent of evaluating a fielded training system. AAI's fidelity verification (FV) model provides an objective rating of the correspondence between the training device and the operational equipment as defined by the required training objectives. The product of all the measured variables becomes the training device fidelity index (FI) and serves to provide a quantitative comparison between the training device (as defined by the training tasks) and the operational equipment that it represents. In the FV model, the training device is evaluated by comparing the required training and operational tasks and the fidelity index is adjusted if additional (i.e., unique) features are included in the device beyond those required by the Task Listings Report. The assumption is that training additional skills may add unnecessary costs that may lead to a loss of device effectiveness.

The FV model allows values to be assigned to basic device characteristics, subdivided into the following categories.

- Task Commonality - Comparison between the operational requirements and the training tasks
- Physical Similarity - Comparison between the physical characteristics required by the training tasks and the simulation work station
- Functional Similarity - Comparison between the functional characteristics required by the training tasks and the simulation work station

JOB: On-Line Maintenance Technician OBSERVER: ______

DUTY: Implementing Preventive Maintenance Procedures

TASK NUMBER	TASK DESCRIPTION	TASK FIDELITY	REFERENCE SOURCE
B.55	Clean UNIBUS expansion box.	(See Note 1.)	(See Note 2.)
B.55.1	Prepare installed unit for cleaning.		
B.55.2	Perform general cleaning.		
B.55.3	Reinstall unit after cleaning.		
B.56	Clean RAMTEK 9460 Graphics Display System (GDS).		
B.56.1	Turn off AC power.		
B.56.2	Open front panel.		
B.56.3	Disconnect all cables from the circuit card assemblies.		
B.56.4	Remove each circuit card.		
B.56.5	Vacuum interior.		
B.56.6	Vacuum circuit cards.		
B.56.7	Wipe down exterior with soft cloth and cleaner.		
B.56.8	Reinstall circuit cards.		
B.56.9	Reattach connectors.		
B.56.10	Close front panel.		

NOTE 1: Document the appropriate cues, required responses, acceptable proficiency levels, response times, etc, for each activity or step.

NOTE 2: Document the reference source from which the task was derived. This may be restricted to the task level and will include document title, number, and date; section identification; and page number. Abbreviations are acceptable if a legend is included.

Figure 2. Tasks Listings Report Format

Values assigned to these categories are combined to produce the fidelity index of each training device. The information required to perform an FV analysis includes:

- List of hands-on tasks/subtasks to be trained (derived from the Tasks Listings Report)
- Description of the controls and indicators used to perform the tasks/subtasks in the operational setting
- Description of the controls and indicators in the training device

Each of the categories cited above individually provide important information on the development of operational training devices. Collectively, however, they provide crucial data early in the design process, which facilitates the verification of critical fidelity concerns. For example, physical fidelity of panels can be verified by inspection during drawing preparation and again after manufacture. Functional fidelity, while identified during the front end analysis, cannot be fully verified until qualification testing but can be confirmed during system and design reviews. The application of this verification process early in the development stage serves to lower the risk and cost associated with the total training program.

Task Commonality (TC) Analysis

The task commonality (TC) analysis in the FV model determines a value for each task by rating whether or not operational tasks that require training are covered on the training device (1 = covered, 0 = not covered). The TC value for a duty is calculated by adding all task (T) ratings within that duty and dividing this sum by the total number of required tasks (TR_t) as determined from the approved Tasks Listings Report. During the TC analysis, subtasks are evaluated in the same manner as higher order tasks. For example, a task with three subtasks would be viewed as three separate items and could receive a maximum rating of 3 (subtask 1 = 1, subtask 2 = 1, and subtask 3 = 1) instead of a maximum rating of 1 as if it were a

single task; therefore, in this model, the terms task and subtask are used interchangeably.

The TC formula is as follows:

$$TC = \frac{T_1 + T_2 + T_n}{TR_t}$$

The task rating scale is defined as follows:

1 = Training device does allow practice of the operational task element.

0 = Particular task element is not represented in the training device.

Physical Similarity (PS) Analysis

In the physical similarity (PS) analysis, the controls and indicators on a training device are evaluated in terms of their size, shape, color, etc, with respect to the operational equipment. In other words, each control and indicator required of the training device is rated against the degree of physical similarity between it and the corresponding control or indicator on the operational equipment. The rating scale used for this purpose is 0 (missing), 1 (dissimilar), 2 (similar), and 3 (identical).

In order to derive a PS index for each task, the ratings given to the physical controls and displays (PCD's) on a device are totaled. This sum is then divided by a combination of 3 times the total number of required controls and displays (RCD_t) plus the number of unique controls and displays (UCD_t). The unique controls and displays on a device are those represented on the trainer but are not associated with the individual task/subtask or activity being evaluated. Thus, the resulting index varies between 0 and 1.00, representing the physical similarity, adjusted for extra or unique features.

The PS formula is shown as follows:

$$PS = \frac{PCD_1 + PCD_2 + PCD_n}{3\ (RCD_t + UCD_t)}$$

The PCD rating scale is defined as follows:

3 = Identical. The trainee would not notice a difference between the training device control or indicator and the operational control or indicator when he or she moves from the training environment to the job situation. Include for consideration the location, appearance, feel, and any other physical characteristics.

2 = Similar. There would be a small noticeable difference for the trainee between the training device control or indicator and the operational control or indicator, but he or she would be able to perform the task. There might be a decrement in performance, but any such decrement would be small and readily overcome.

1 = Dissimilar. There would be a large, noticeable difference, quite apparent to the trainee, between the training device control or indicator and the operational equipment and a large decrement, given that the trainee could perform at all. Specific instruction and practice would be required on the operational equipment after practice on the training device to overcome the decrement.

0 = Missing. The control or indicator is not represented at all in the training device.

Functional Similarity (FS) Analysis

The functional similarity (FS) analysis compares the controls and indicators of a training device to those in the operational equipment in terms of amount of information conveyed from or to the human operator. Just as in the PS analysis, each of the required controls or indicators relevant to a particular task/subtask receives a rating from 0 (missing) to 3 (identical).

In order to calculate the FS index for each task, the ratings given to all functional controls and displays (FCD's) on a device are summed and the total is divided by a combination of 3 times the total number of required controls and displays (RCD_t) plus the unique controls and displays (UCD_t). This results in an index ranging from 0 to 1.00.

The FS formula is shown as follows:

$$FS = \frac{FCD_1 + FCD_2 + FCD_n}{3\ (RCD_t + UCD_t)}$$

The FCD rating scale is defined as follows:

3 = Identical. The number of states in the training situation is the same as the number of states in the operational setting (as defined by the training tasks).

2 = Similar. The number of states in the training situation is at least 75 percent of the number of states in the operational setting (as defined by the training tasks).

1 = Dissimilar. The number of states in the training situation is less than 75 percent of the number of states in the operational setting (as defined by the training tasks).

0 = Missing. The control or indicator is not represented at all in the training device.

COMPUTING DEVICE FIDELITY USING THE FV MODEL

Just as the identification of fidelity requirements are at the task level, so too is the application of the FV model. Each task is analyzed and quantitatively evaluated against the operational equipment as previously discussed.

Again, using the example previously cited, the FV values for task B.55 may be seen as TC = 1.00 and FS = 1.00 (the assumption in this example is that each task/subtask and tactile and functional fidelity is identical to that of the operational environment).

FC values are computed in a similar manner for each task within each duty category. Consequently, values are totaled at the duty level and FV values can be extracted for task commonality and physical and functional fidelity. Just as the identification of fidelity requirements was gathered from each duty category and used to define the simulation work station requirements, the duty FV values are gathered and a device fidelity index (FI) is derived.

The FV Fidelity Worksheet is shown on Figure 3 with task B.55 entered to illustrate the application of the FV model. Worksheets are normally separated by duty category and FV values assigned after all the tasks within the duty category have been analyzed (it is a simple matter to isolate individual tasks within the duty category and compute the FV values separately). The computational (lower) portion of the worksheet would appear only on the last page of each duty category being evaluated. This organization easily facilitates the computation of the FV index for each duty category.

On Figure 3, the total value of the tasks (tasks and subtasks) is assumed to be 65; there are 65 required tasks on the device; the physical similarity total value is 144 with 50 required controls/displays and 15 unique controls/displays; and the functional similarity total value is 162 with 65 required control/display actions and no unique control/display actions. The values for this duty, within the previously cited parameters, are TC = 1.00, PS = 0.74, and FS = 0.83.

The analyses just presented are used to calculate an overall fidelity index for each duty. The total values for TC, PS, and FS are summed and divided by 3. This value represents the overall degree of correspondence between the training device and the operational equipment, as defined by the training objectives, for the duty being evaluated. The fidelity index (FI) value for each duty will range from 0 to 1.00. This value is based on the premise that the higher the level of fidelity (1.00 or 100 percent) the more effective the training device; as the level of fidelity decreases, there will be a corresponding decrease in its ability to teach course objectives. In the prior example, the TC value of 1.00, PS value of 0.74, and FS value of 0.83 are combined to yield a total fidelity index (FI) of 0.86 (1.00 + 0.74 + 0.83 divided by 3) for duty B.

Therefore, for duty B of the hypothetical simulator used in explaining the fidelity verification model, the training device is 86 percent (FI = 0.86) identical to the actual operational equipment (tasks and tactile and functional fidelity) as defined by the lowest order learning activities. In this example, the 100-percent fidelity requirement was not met (i.e., FI = 0.86); consequently further analysis is required to identify the deficient task(s) and to initiate corrective action. The FV model supports this analysis and allows individual components (tasks/activities) to be isolated for further investigation and modification. At the job (simulation work station) level, the FV values for each duty can be gathered and averaged to provide a figure of device quality. It should be remembered, however, that the fidelity issues are identified and resolved at the lowest levels (duty, task, and activity).

IMPLEMENTING THE FV MODEL

The FV model, in consonance with the emerging Task Listings Report, will initially be applied to all early system design discussions. As the front end analysis reaches maturity, the individual fidelity issues will redefine system parameters and the FV model will provide both contractor and Government personnel with a vehicle for verifying fidelity accuracy.

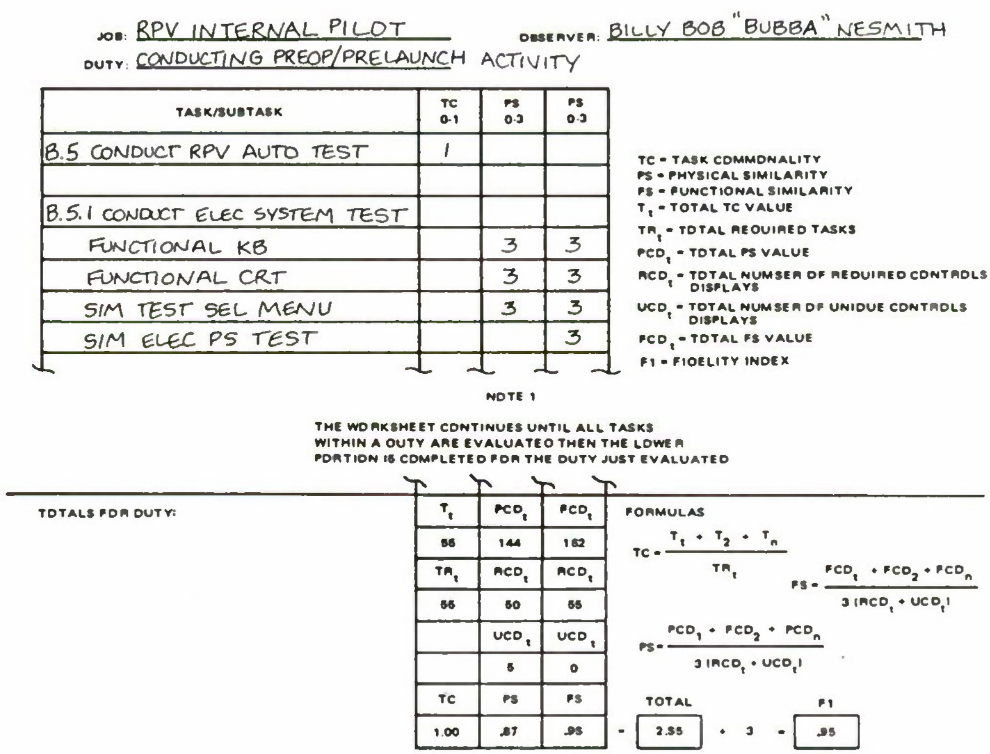


Figure 3. Fidelity Worksheet

Initially used as an internal tool, the FV model becomes a paper verification of fidelity definition at the Preliminary Design Review (PDR) and again at the Critical Design Review (CDR). The baseline data documented at the lowest training level (tasks) can then be evaluated objectively throughout the development, production, and testing process. For example, if at PDR, a fidelity discrepancy is revealed in design strategy relative to a specific training task, it can easily be reviewed again at CDR and ultimately be evaluated as a separate entity during system testing.

VERIFICATION OF FIDELITY

Test plans and procedures were developed to verify the physical and functional requirements of the simulation work stations. The identification of fidelity requirements between the operational equipment and the training system will have been well defined by this point, and the individual work stations will be ready for system level testing.

Physical verification of the work stations has been an ongoing process throughout the design and layout of hardware components. This verification included such considerations as size, location, and color of controls; readability of displays; and accessibility of components that would require service or maintenance. In addition, safety was also evaluated during physical verification to identify, and control, potential hazards to personnel and equipment inherent in system hardware.

Functional verification, at least on paper, has also been an ongoing consideration during the prior design and development process. Hardware/software interactions and man-machine interface, as documented in the Tasks Listings Report, have been constantly satisfied up to this point.

Verification of fidelity requirements will be accomplished primarily through the application of inspection and demonstration procedures, although analysis and testing may also be justified at some later point in the process.

By definition, inspection is the verification that a specification requirement has been met by visual examination of the item, review of descriptive documents, and comparison to a deliverable product of specified standard. In reality, the inspection of each student work station will confirm that all required levels of physical fidelity are present to ensure the direct transfer of each training objective into the work world. In other words, 100-percent physical fidelity will be represented in each simulation work station as determined by the training tasks.

Functional fidelity, on the other hand, is verified by exercising hardware and software under simulated operational conditions for visual confirmation that fidelity requirements are met. It will be demonstrated that system-generated cues will elicit structured or free play inputs and, in turn, the system will react realistically to the students' activities as defined by the previously identified training requirements.

Acceptance or rejection criteria is determined as a GO/NO GO or PASS/FAIL evaluation. Acceptance is defined as satisfying 100 percent of the fidelity requirements identified in each learning objective, whereas rejection is defined as anything less than 100-percent achievement.

A simulation work station will be considered acceptable only after each duty is acceptable, then only if all tasks within that duty have met the fidelity requirements specified for training. In the event that a task fails to satisfy its defined fidelity requirements, that task is rejected and the duty (and consequently, the work station) is unacceptable until the level of fidelity required to teach that task is corrected and the task is accepted. With this approach, each simulation work station will then be 100 percent responsive to the required fidelity level necessary to teach operator, maintenance, and analyst personnel to perform their jobs when they report to their future duty assignments.

CONCLUSIONS

Simulator fidelity, as a concern, seems to occur completely in an ex post facto environment. For example, the normal sequence of events is to build the trainer, train the students, send them to the field, then conduct a study to determine trainer effectiveness. AAI submits that this is too late. Trainer effectiveness must be a primary concern at the outset and must be controlled throughout the entire design and production process. This can only be done if system design parameters are viewed simultaneously with the identification of training tasks. AAI's fidelity verification (FV) model is one such approach. AAI accepts the fact that the FV model is in its infancy and requires much more data, but holds fast to the idea that the FV approach, or some other model, is critical if the job analysis process is to have its full impact on the emerging training system.

REFERENCES

Hays, R.T. (1981, November). Research issues in the determination of simulator fidelity: Proceedings of the ARI sponsored workshop. (Technical Report No. 547). Alexandria, VA: US Army Research Institute for Behavioral and Social Sciences.

Massey, R.H. (1986, March). Taxonomic considerations in the acquisition of maintenance training simulators: Proceedings of the Air Force Conference on Technology in Training and Education (TITE). Montgomery, AL: pages V-146 - V-150.

Rouse, W.B. (1982). A mixed-fidelity approach to technical training. Journal of Educational Technology System, II (2).

Tufano, D.R. and Evans, R.A. (1982, April). The prediction of training device effectiveness: A review of Army models. (Technical Report No. 613). Alexandria, VA: US Army Research Institute for Behavioral and Social Sciences.

ABOUT THE AUTHOR

Dr. Bruce is a Senior Engineering Psychologist in the Logistics Division of AAI Corporation. He holds an Ed.D. in Higher Education and has 15 years experience in the design and development of instructional training systems. Since joining AAI in 1984, he has been the ISD Manager for various Government contracts leading to the design of full fidelity SIGINT/EW training devices.

DESIGNING TRAINING DEVICES: THE OPTIMIZATION OF SIMULATION-BASED TRAINING SYSTEMS*

Michael J. Singer, Ph.D.
Army Research Institute
Orlando Field Unit
Attn: PM TRADE/PERI-IF
Naval Training Systems Center
Orlando, FL 32813-7100

and

Paul J. Sticha, Ph.D.
Human Resources Research Organization
1100 South Washington St.
Alexandria, VA 22314

ABSTRACT

Effective training devices are those that meet training requirements at minimum cost, or provide the maximum training benefit for a given cost. The Optimization of Simulation-Based Training Systems (OSBATS) is a model that is designed to facilitate the investigation of tradeoffs involved in developing effective training device concepts. The model is based on benefit and cost approximations that are used to analyze tradeoffs between various training device features in developing a device configuration, and then conducts similar tradeoffs between different training device configurations. The development of OSBATS has been more theoretical than the typical decision support system or aid, but shares many of the attributes of the standard decision aid. The tools or modules that comprise the model address the following activities: a) the clustering of tasks for developing coherent training device configurations, b) the identification of optimal instructional features for a task cluster, c) the specification of optimal fidelity levels for a task cluster, d) the selection of the minimum training device family that meets training requirements, and e) the allocation of training resources in the family of suggested training devices. The final output of the OSBATS model is a functional description of the optimal set of efficient training devices given the tasks, training criteria, and cost constraints.

INTRODUCTION

The development of training systems is a complex undertaking that uses behavioral principles of learning to convey specific content-domain skills and knowledges. Training systems often incorporate training devices that are as complex as (and sometimes more complex than) the actual equipment that they are designed to provide instruction about. A fair amount is known about how training systems should be designed and implemented, and what the varied tradeoffs actually mean in terms of performance, training effectiveness, and overall cost. Within that large complexity is the "smaller" problem of designing a training system strategy based on training devices. Although there is considerable amount of data about specific training devices as used within specific training systems, there is no organized body of information necessary to build effective training device based systems or segments (5). As a result, the design of effective training devices is an effort that is fraught with imperfect data, opinion-based design rules, and an increasingly large number of choices in the large array of technologies that can be used to address any single training problem.

Training Devices

The goal of training device concept formulation is to propose a training device that meets training requirements at minimum cost, or provides the maximum training benefit for a given cost. This fits the spirit of Hall's definition (4) of optimization - "securing the best fit between the system and its environment" (p.73). The approach to training device concept formulation that has always been used in the past is to rely on the experience and knowledge of engineering and education professionals. This process has in no way approximated optimization.

One major problem area for training device developers is the range of information required for the large number of tradeoffs that must be made in order to arrive at a concept for an effective training device. Any tasks identified as requiring a training device solution must be analysed in order to understand and explicitly state the training device configuration required to meet the task

* The views, opinions, and/or findings contained in this report are the authors' and should not be construed as an official Department of the Army or Department of Defense position, policy, or decision, unless so designated by other official documentation.

training goals. This requires that the to-be-learned aspects of the task equipment and environment be identified, the technological options for simulating the necessary aspects of the equipment and its environment must be known, and the cost of using the technology in this particular way be known. The reliability and maintainability of the training device as conceived, the effectiveness of the training device in teaching the requisite skills and knowledges, and how the training device will or should be used are all prime concerns of the training device developer during this process.

A problem with this detailed approach is that when new technology, or improvements arise, the experts must estimate it's effectiveness and attempt to apply the technology appropriately. Individuals involved in training device design are seldom exposed to reliable information about how that applied technology actually works in the training system. The process of developing training devices would thus be improved if there were some way for designers to access and use training device and system evaluations, research experiments, and the combined experience of school professionals.

Design Aiding

The U.S. Army Research Institute for the Behavioral and Social Sciences and the Project Manager for Training Devices have embarked on a research and development program that addresses the problem of training device design. This program is an attempt to organize the large body of training technology and learning theory currently available, and develop an implementable model for aiding training developers in evaluating training device altenatives. The initial effort has been in the development of a model that provides tools for doing tradeoff analyses of training device configuration concepts. The prime contractor in this effort has been the Human Resources Research Organization (HumRRO). Together we have developed a model named the Optimization of Simulation-Based Training Systems (OSBATS). The OSBATS system is computer based and is structured to use databases and rule-based procedures interactively during the training device concept development process. Given the users choice of constraints and assumptions, different users may develop different solutions for the same problem, but the differences will be based in design rationales and have a supporting audit trail automatically provided.

The OSBATS tools are designed to aid the developer in providing an answer to the question "how much simulation is enough?" The tools have been developed by taking an innovative, theoretically based, top-down analytical approach. Army tasks were selected as a basis for analysis, since that information was more readily available. Future efforts will focus on detailing the data at the skill and knowledge level, so that model information will be more robust in application. The central question is need for simulation (ie. fidelity) versus the cost of providing the appropriate levels of simulation, hence the central module is a tool for Fidelity Optimization. Training effectiveness is also influenced by the instructional basis, and instructional features also have a significant effect on the cost of the training device, which led to the inclusion of an Instructional Features tool. The fidelity and instructional features modules approaches work best when the tasks form a coherent cluster of simulation needs. This led us to a Simulation Configuration tool, which clusters tasks in terms of simulation requirements and fidelity based cost estimates.

The problem of coherent training device design has another major factor, separate from instructional considerations. The cost of developing and using the training device must be considered in order to be efficient in training. Cost is driven by the time required to train each task on the training device. The concept formulation process must ensure that the minimum family of devices for the tasks are developed, and the Training Device Selection tool serves this purpose. Time in training programs is also limited, and constraints are imposed by student flow. These factors and the training plan help determine the numbers of training devices required, which in turn effects training program resources. The Resource Allocation tool estimates the number of devices needed to meet requirements, working to derive the optimal family of training devices for the tasks, training resources, and student flow.

A few more details about what we mean by optimization are necessary to further explain our approach. In terms of training devices, an optimal choice is one that returns the most for the least, meaning devices that produce the greatest gain in student skill and knowledge for the time in training, the investment expense, and the operating cost. The identification and prediction of gains in skills and knowledges is not a trivial data collection problem. The gain or benefit can be an increase in the amount learned, an increase in proficiency, or a decrease in the time required to learn a set amount of information or reach a set level of proficiency. The realm of resources presents another hard data problem in attempting to optimize training devices. We are attempting to cover the greatest part of these resource areas by including all development, maintenance and overhead cost into one measure. The number of students, number of devices required, and minimal family of training devices for training the task set are also considered, but separately from cost.

DECISION SUPPORT

The OSBATS is a decision support system or decision aid that should increase the timeliness, amount, and quality of information available for Army decision makers during the training device concept formulation process. As Sprague and Watson (9) have pointed out, all decision support systems are composed of three basic parts or subsystems: the data that the system uses, the user interface, and the decision models that use the data to recommend decisions. The three subsystems for OSBATS are briefly discussed in the next few paragraphs.

User Interface

The user interface is a critical part of any decision aid, and serves as the basis for user understanding and confidence in system processes and recommendations. The OSBATS model is meant for use by engineering and educational professionals involved in training device concept formulation efforts at the office of the Project Manager for Training Devices (PM TRADE). Obviously the more naive a user is the simpler the interface must be. Also, the more a user knows about the process, the more justification for recommendations are needed, along with shortcuts for situations where the user is satisfied with the system. Users are also considerably interested in being able to manipulate the system, to explore different problem options. The OSBATS system attempts to deal with a wide range of users through the use of graphs and tables to present results of the tradeoff analyses performed and the information used in making the analyses. This provides the user with different ways of viewing the results. The user inspects and modifies the information by using a mouse activated set of commands and selections. The results are inspected along with the data and reasoning used in the analyses by using the same interface. The system is also being modified in order to produce output data files or simple printed reports of the results of the analyses.

Database

The data subsystem required for decision aids usually consists of a database of information; procedures for collecting, organizing, and entering the data; and an inquiry or retrieval system for accessing the data. There are two ongoing contractual efforts involved in developing various aspects of the required data subsystem (2, 3). The goals of these efforts are to detail the internal data and rules required for the models; to describe the input data required for users to initiate work with the models; to identify or develop methods for collecting, converting and/or transforming the data to model usable formats; and to define the necessary framework for organizing the varied rules and data required by the optimization efforts.

As indicated above, there are two types of data required to support the functioning of the model tools. The first type of data, called resident or internal data, covers the unchanging or slowly changing information and relational rules involved in the generation of options, tradeoffs, and configurations. The second type of data required by the models is situationally specific data, the data used to initiate execution of the models.

The resident or internal data cover general task characteristic based rules for fidelity options, types of instructional features, fidelity and instructional feature cost estimates, learning parameters, and so forth. These data and rules will be developed through analytical evaluations and data collection efforts, including experiments designed to verify certain assumptions and the hypothesized relationships within the model. The resident data include rules about the relationships between the resident data values and the input data. These resident data will be available to the OSBATS system through a modifiable data base system.

The situationally specific or input data are used to initiate execution of the models. These data include descriptions of the tasks to be taught, the task performance criteria to be met by the training, the current training investment and operating cost projections, the type of instructional approach, number of students, number of instructors, time for training each task, etc. These data should come from the analysis of training requirements conducted during the development of the program of instruction.

The data collection work currently underway is focused on the resident data and includes detailing the data required for the models, planning for and acquiring the rotary wing operations task data, and developing a prototype database system for the resident data. It should be made clear that the resident data are related to the input data in terms of the descriptive task variables used, such as standards, conditions, equipment, cognitive and psychomotor classifications, criticality of performance, etc. These data must be acquired from many sources. The resident database uses these variables within rules that specify applicable fidelity dimensions, fidelity levels, and instructional features for specific tasks. These rules must have explicitly defined task variables in order to be structured for general use across tasks. For example, a simplified fidelity rule about how much platform motion to include (a fidelity output variable) might require information about the entry level proficiency of the student and the degree to which kinesthetic motion cues are used in guiding task performance (two variables and associated values). This forms the basis of the internal structure of the resident rule, and directly specifies the types of variables and values to be collected for an analysis session. In this way the internal

resident data structure must be linked to the structure of the input data. The system would not be able to make a recommendation on the degree of motion unless the required input data for the rule were provided.

As discussed above, the domain of training device design requires reliable information about applied technology and the implications for training systems. Hence the resident decision rules and models come from varied sources, including psychological experiments and theory, training system evaluations and validations, and subject matter expert opinion. The primary problem in this approach is the development of a reasonable framework, in addition to the expense and time required in organizing explicit information into a usable format. With the approach we have described here, we believe that a workable framework has been developed.

Models

The central tools within OSBATS are those that focus on specific instructional features and specific levels on identified dimensions of fidelity. The goal of the system is to prescribe a training device configuration that has the greatest benefit for the projected cost. The benefits are either experimentally based with reference to transfer of training or are estimated by experts. The costs used include the investment and operating cost of the training device over it's life cycle. As introduced above, the model consists of five conceptual tools:

1) Simulation Configuration Module - a tool that develops clusters of tasks sorted into the categories of part-mission training devices, simulators, and actual equipment.

2) Instructional Feature Selection Module - a tool that analyses the instructional features needed for a set of tasks and specifies the optimal order for user selection of instructional features.

3) Fidelity Optimization - a tool that analyses the set of fidelity dimensions and levels, then provides the optimal order for inclusion of fidelity dimensions and levels given the task set.

4) Training Device Selection - a tool that aids in determining the most efficient family of training devices for the entire task group, given the training device fidelity and instructional feature configurations developed.

5) Resource Allocation - a tool that aids in determining the optimal allocation of training time and number of training devices needed in the family of training devices recommended.

The OSBATS model was developed as a framework that allows the addition and insertion of new models for different aspects of the concept formulation process. Each of the five areas was identified through the analysis of the theoretical basis of the training device concept formulation process. Each of these modules is based on empirical information, assumptions, and hypothesized relationships. Each of the modules uses training system and task data to present options, tradeoffs, and recommended configurations to the users.

Fidelity in Simulation. The Fidelity Optimization module currently requires input data about the cue and response requirements of each task in the task set, in order to match those requirements to the appropriate fidelity dimensions. This supports the analysis of the highest cost drivers in the development of a training device, by selecting only the specific dimensions that are needed. The module then uses the cue and response information to select the optimal family of dimensions and levels based on the tradeoff between the benefit to training provided by each level and the cost of developing that level of fidelity. The model will evolve to include other cost aspects such as maintenance and life cycle operating costs.

The model functions by first calculating the benefit to cost ratio for each level in each dimension, then using a selection process to arrive at the most effective combination of fidelity dimension levels. There are two ways for the user to specify what is the "most effective" combination. One way is to determine the level of funding available for the training device; the model then identifies the most beneficial dimensions and levels for the task set at that level of funding. Another way is to specify the benefit (which represents training effectiveness) desired in training the task set, and let the module select the fidelity dimensions and levels needed to reach that degree of benefit.

The fidelity module provides several methods that enable the user to conduct "what if" analyses. One method is to restructure the task set by changing the tasks that are included for analysis. The user can eliminate a task that is driving a high level of fidelity along one or more dimensions, possibly arriving at a configuration that meets all of the fidelity requirements for the reduced task set for the projected available dollars. This recommendation would serve as a basis for discussions with the school about the need for more money to train particular tasks, or the need for restructuring the way the tasks are trained at the school. Another method for analysis is provided by reducing the levels within any fidelity dimension. This feature allows the user to force a higher level of some fidelity dimension at the start, or preclude a level from consideration. Finally, the user can use the feature to eliminate a fidelity dimension entirely. This allows the user to study what levels of benefit might be achieved for the same money on the other dimensions required for the tasks.

Instructional Feature Selection. The Instructional Feature Selection Module is used to select features that will improve the efficiency of training on the proposed training device. The module uses input data such as task training requirements and projected costs for task training on actual equipment. The module also uses internally resident cost data and applicability rules to select the features relevant to the task set, assign costs and calculate the benefit values for the each task and feature match. The benefit values are summed across all of the tasks to which they apply to arrive at a total feature benefit value. The benefit and cost are then combined as a ratio that provides a measure of how much increased efficiency can be acquired for the dollars spent. This ratio is used to order the instructional features along a curve. As with the fidelity module the selection method allows the user to choose the most beneficial instructional features for a constrained dollar amount, or to select a set of instructional features that provide the greatest proportion of benefit for the task set at the lowest cost.

The user can also conduct "what if" studies using the same functions as are available in the fidelity module. The user can restructure the task list under consideration by including or excluding tasks that require multiple instructional features or single, expensive instructional features. The user can also eliminate instructional features from consideration, which could change the ordering of suggested features. Finally, the user can force one or more instructional features to be automatically included in the package, which might show decreases in the optimality of the constrained instructional feature order.

FUTURE PROGRESS

The pre-prototype tools or modules have been individually developed and evaluated for their user interface and theoretical foundations during the winter and spring of 1987. The integrated prototype system has been delivered for evaluation of the complete model, and will be revised to increase flexibility and broaden applicability during the next year.

The major problem in this detailed approach to instructional features and fidelity is that there is very little or no available experimentally based information in the literature that has focused explicitly on the interaction of task types and fidelity (1). Some ARI theoretical efforts in the past have focused on the identification and characterization of training system and training device variables (5, 6, 8), but the amount of correctly structured information is limited. On the basis of this earlier theoretical work and a concept demonstration (8, 6), it does seem possible to structure empirical knowledge about the characteristics and instructional features of training devices in production rule formats. This requires careful analysis of the gross level training device evaluation reports that are available, in conjunction with the better research (e.g. 7).

The next version of the OSBATS model will borrow from expert system technology in order to incorporate processes that infer the requirements for instructional features and fidelity specifications from more basic information about the tasks being considered. The inference process will be implemented through a commercial authoring tool that supports a production rule architecture. The output of the production system will provide the input parameters required by the analytical portion of the OSBATS model. The higher level analytical routines and user interface functions will be retained from the current version of OSBATS. The next version of OSBATS will also contain direct connections to the commercial database being prototyped in the Database Development (3) and Data Collection (2) contracts.

The use of an expert system authoring tool and production-rule approach should allow for the collection of more accurate data without increasing the workload of the subject-matter experts who must provide many of the judgments required by the model. In addition it provides a framework for encoding what is known from the research literature. Finally, the format will allow for incremental growth of the model, accommodating future empirical and fundamental research.

Because the domain specific information used by OSBATS (e.g. fidelity dimensions and levels, instructional features) is represented in simple knowledge base files, new domains can be incorporated into the system without changing the functioning of the model. This will allow us to expand the prototype OSBATS from the current rotary-wing operations domain to other domains (for example, armored vehicle operations or electronic maintenance domains). It will also allow the incorporation of new modules (such an the instructor operator considerations module now under conceptual development). We are currently planning for the extension into another domain, and will be testing the practicality of the approach during FY 1988.

GENERALIZATIONS AND IMPLICATIONS

OSBATS is a theoretically based model that trades off the projected benefit of discrete features relevant to specific tasks against the cost of developing and fielding that combination of features. OSBATS is a flexible rule based system that will use expert system technology to represent what is known or surmised about the benefit of features and aspects of training devices in relation to specific tasks. OSBATS is an expert system based decision aid that doesn't model any single expert.

OSBATS represents the current movement into the development of decision aids that are designed to expand the number of factors that the average decision-maker considers, while increasing the speed with which decisions are made. The typical training device designer considers many factors in general ways during the development of a training device concept. Different training device designers consider different subsets of those factors. OSBATS and decision aids like OSBATS (in other application areas) support the decision maker in consistently considering as many aspects as can be identified, by using information from research and other experts. This decreases the individual-to-individual variance in the number and type of aspects considered in comparable cases. Decision aids like OSBATS also serve to increase the shared information base about the tradeoffs to be made, and can serve to increase the consistency of the decisions that are made. This is true even though the decision aids can be used on the same problem by two different users to develop two different approaches. The last great benefit for using decision aids like OSBATS is that the reasons for those differences are immediately present in the audit trail of user decisions that are made during the decision aid sessions.

Perhaps the most important point is that decision aids like OSBATS serve to identify what is not well known in particular domains. In codifying the research literature and the consistent experientially based knowledge of experts, information weak areas are identified that do not have any firm available answers. Many of these areas are known to researchers, although perhaps not all, but the primary benefit of the organizational process is to specify exactly what is known, what must be assumed, and provide a rough measure for prioritizing what should be investigated next.

REFERENCES

1. Ayres, R., Hays, R. T., Singer, M. J., & Heinicke, M., An Annotated Bibliography of Abstracts on the Use of Simulators in Technical Training. (RP No. 84-21), U.S. Army Research Institute for the Behavioral and Social Sciences, Alexandria VA, 1984.

2. Data Collection in Support of the Optimization of Simulation-Based Training Systems, Delivery Order 0002, contract N61339-86-D-0009.

3. Database Development in Support of the Optimization of Simulation-Based Training Systems, contract MDA-903-86-C-0382, University of Central Florida Subcontract 872830.

4. Hall, A. D., A Methodology for Systems Engineering. Van Nostrand Reinhold Co., N.Y., 1962.

5. Hays, R. T., & Singer, M. J., Research Issues in Training Device Design: The Organization of a Data Base. (TR 588), U.S. Army Research Institute for the Behavioral and Social Sciences, Alexandria, VA, 1983.

6. Hays, R. T., & Singer, M. J., Training System Design Guidance: Accumulating and Using Empirical Data. North Atlantic Treaty Organization Defense Research Group Panel VIII Symposium, "Transfer of Training to Military Operational Systems," Brussels, Belgium, October, 1985.

7. Rouse, W. B., Rouse, S. H., Hunt, R. M., Johnson, W. B., & Pelligrino, S. J., Human Decision-Making in Computer-Aided Fault Diagnosis. (TR 434), U.S. Army Research Institute for the Behavioral and Social Sciences, Alexandria, VA, 1980.

8. Singer, M. J., & Perez, R. S., A Demonstration of an Expert System for Training Device Design. Journal of Computer-Based Instruction, Vol. 13, No. 2, pp. 58-61, Spring 1986.

9. Sprague, R. H., & Watson, H. J., MIS concepts Part I, in W. C. House (Ed.), Interactive Decision Oriented Data Base Systems. Mason/Charter Publishers, Inc., New York, 1977.

ABOUT THE AUTHORS

Dr. Michael J. Singer is a Research Psychologist with the U.S. Army Research Institute for the Behavioral and Social Sciences at the Orlando Field Unit. His Ph. D. is in Cognitive Psychology from the University of Maryland. His work has primarily been in training and simulation with an emphasis on fidelity in the training situation. His interests also include memory, cognitive processes, mental modeling, judgment, and decision-making.

Dr. Paul J. Sticha is a Senior Staff Scientist with the Human Resources Research Organization in Alexandria, Virginia. He was formerly with Decisions and Designs, Inc., as a Senior Decision Analyst. He received a Ph. D. in Mathematical Psychology from the University of Michigan. His current interests are in the development of computer-based decision aids, mathematical models of human performance, applied decision analysis, and training technology.

TRAINING SYSTEMS R&D PROGRAM: PROGRESS AND CHALLENGES

Dr. Ronald Hofer
PM TRADE, Orlando, FL 32813

and

Dr. Halim Ozkaptan
Dr. J. Peter Kincaid
Army Research Institute, Orlando, FL 32813

ABSTRACT

The training device and simulation community has achieved the technological power to simulate military systems and operations with impressive realism. This technological strength is offset by the fact that we do not always consider the cost and potential training benefits of alternative approaches, and the training effectiveness of the training systems that we field.

This paper describes a joint R&D program between the Army Research Institute (ARI) and the Program Manager for Training Devices (PM TRADE) to provide training developers and engineers a set of tools to establish the capability for evaluating training alternatives with respect to: (1) desired effectiveness at minimum cost, or (2) maximum effectiveness at a given cost. We are developing computerized decision aids with supporting databases and procedures to help optimize the training development process.

The program upon which we have embarked addresses: (1) the implications of MANPRINT for developing simulator/device based training systems, (2) the analysis of training requirements to determine skills and knowledges to be trained ; (3) the development of training strategies, (4) the question of how much simulation or fidelity is enough given that a training device or simulator is needed; and (5) the best manner of implementing embedded training. We are also examining optimal ways to organize and present the information needed for embedded training and electronically presented technical information.

INTRODUCTION

As witnessed by this conference, and the development of sophisticated and complex training systems, the technology of training devices and simulators continues to develop explosively. The engineering community has the capability to simulate with impressive fidelity our major weapon systems. Despite this growth and the level of sophistication which we have achieved, there remain many challenges. We would like to present some of these challenges as well as describe the course of research we have set to develop new tools and capabilities which lead to their solution. While we speak from the Army perspective, we believe these challenges are true for our sister services as well.

OVERALL CHALLENGE

The following represent what we see as our key challenges today:

o <u>Development of Comprehensive Training Solutions</u>

While the industry-government team has been skilled in developing sophisticated training devices and simulators, these devices and simulators have not been developed within the context of comprehensive training systems. Our devices are not planned as part of a comprehensive training solution to assure cost-effective and responsive training. We need to develop training strategy based devices rather than device based training strategies relative to well defined performance criteria.

o <u>How Much is Enough?</u>

Given the need for a training device or simulator, the answer to one question has eluded us. How much simulation and fidelity do we need to satisfy the training requirement? Is a multi-million dollar simulator needed or will a much less complex simulator be adequate? Will a simple training device do the job? We are all aware of the range in sophistication and cost of training alternatives for the same requirement.

o <u>Disciplining the Design Process</u>

The results of training and cost implications of alternative approaches are not fully assessed or considered. Our solutions are not constrained by cost or performance requirements. We must become more concerned about the <u>affordability</u> of training solutions, and <u>development</u> of long term investment strategies for implementing these solutions. A dollar constraint would invariably result in lower cost approaches. A required performance outcome should lead to new and perhaps innovative solutions. A requirement to meet a performance standard within a

prescribed period of time would also pose a significant challenge. In addition, we often do not require a design rationale or audit trail to justify the design approach adopted. Alternative design concepts are encouraged, with evaluation criteria and measures of effectiveness to guide selection of the optimum approach relative to cost and effectiveness.

o Results of Training

On the other hand, no hard analysis of training alternatives is possible unless we can measure the results of training. We need performance data to assess the relative training value of alternative approaches. For example, what is the relative training value of a Video Disc Gunnery System (VIGS) versus a Unit Conduct of Fire Trainers (UCOFT)? Despite the cost of training systems, no comprehensive assessment technology or performance assessment program are in place.

o Adding Training Value

Simulation or replication of the operational equipment does not assure training. The relationship between behavioral requirements, learning theories, training media, student aptitude levels and instructional processes are not well understood. In this respect, the difference between training devices and simulators is often overlooked. From our perspective, a training simulator represents a replication or "analog" of the weapon systems being addressed. A training device, on the other hand, can be likened to an "analytical" model, whereby explanatory principles are demonstrated or enunciated. Embedded training sharpens the focus on this issue, and may be more suited to an analog model. We must break the mental set of striving for a simulator which provides practice, at the expense of not providing an understanding and insight into the processes being replicated and the relevance to combat readiness.

o Training to Fight

Training to operate is not the same as training to fight. This distinction is not fully recognized or accommodated. SIMNET technology has made important strides in providing a capability for training to fight. The results of stress and fatigue on performance must also be understood to assure the necessary overtraining, cross-training and other requirements to overcome performance decrement.

o Timely Development

The timely development of training systems parallel to the weapon system development process has eluded us. The MANPRINT initiative together with embedded training, requires early conceptual description of soldier system interface and job performance requirements. It also requires the combat developer and training developer to develop integrated Operational and Organizational (O&O) concepts which express strategies, performance requirements and envisioned usage for both weapon and training systems. We must meet these goals to assure timely development.

o Presentation of Information

Embedded training, portable electronic maintenance aids and videodisc training devices have opened a new realm of how to organize and present information, and how to prepare (author) such information. These new technologies offer an opportunity to by-pass technical manuals and to prepare the necessary information in a more effective and less costly manner.

ARI and PM TRADE have joined forces to collectively address some of the above issues. The purpose of this paper is to describe our initial and emerging efforts.

PROGRAM GOALS

Our primary goal is to permit the evaluation of training alternatives with respect to: (1) desired effectiveness at minimum cost, or (2) maximum effectiveness at a given cost. Our approach to achieve this goal is to develop a computer based system with supporting databases and procedures to permit interactive utilization by multi disciplined teams for the support of the training development process.

We are obviously not starting at the beginning. A large amount of information and technology is available. What is new, is our attempt to organize, in a comprehensive and systematic way, the large body of training technology and information now available in a manner which addresses the challenges set forth earlier. We are relying on the power of our new computer systems and networks to implement the required analytical and analog models, and their supporting databases. We envision a sufficiently flexible system so that different users may come up with different solutions to the same problem. However, they will be able to provide a design rationale and audit trail for the decisions that they have made.

Today we will describe our interrelated program objectives, and some of our more significant accomplishments. We will start with our effort to interface with early weapon system development through the MANPRINT initiative. During the conceptual phase of weapon system development it is important to assess the impact of different design concepts/alternatives on training requirements. This consideration is a key part of MANPRINT objectives to assure reasonable and achievable manpower, personnel and training demands of emerging weapon systems. In addition to assessing the training impact of a design

alternative, we must also identify at this time which portions of the training requirement should initially be allocated to embedded training. As part of this effort, we are developing techniques to: (1) evaluate the impact of different weapon system design concepts on training requirements, (2) assess the costs of potential training systems needed to meet these requirements, and (3) identify early candidates for embedded training.

Our second major effort addresses the formal training development process. It is designed to conduct the necessary front end analyses of a training requirement, in order to provide the necessary input for the development of a training strategy or system. Training strategies are needed during the materiel development cycle to put embedded training, and other training approaches in context, for parallel development with weapon system development. In addition, the new Army policy to acquire weapon systems as families, such as the Army Family of Vehicles and the Light Helicopter Experimental (LHX), requires the early development of a training strategy. This effort includes: (1) how a training requirement should be stated to support a comprehensive training requirements analysis and an effective training strategy, (2) the development of computer based aids for the analysis of training requirements to provide the necessary input data for the development of training strategies or training devices/simulators, and (3) methodology for the development of training strategies.

Our third effort deals with the Optimization of Simulation Based Training Systems (OSBATS) which addresses the cost-effective design of training devices and/or simulators. OSBATS is a family of computer based models designed to determine how much simulation is enough during the concept formulation process.

The above efforts are supported by two major database efforts. The first database, identified as "Functions and Tasks," is being designed to support the analysis of training requirements, and will in part rely on MANPRINT Data. The second database, identified as "Resident Data" will support the data internal to the optimization models, to be processed by their rules and algorithms. These efforts are being supported by the OSD Training Performance and Data Center.

Additional efforts described in this paper include: (1) embedded training, with particular emphasis on criteria for the utilization of embedded training, and the necessary design trade-offs to assure cost-effective training and (2) authoring efforts, which are designed to support material needed by embedded training, portable maintenance aids (e.g., MEIDS), and electronic classroom training aids (e.g., EIDS).

MANPRINT INITIATIVES

During the conceptual phase of weapon system development, it is important to assess the impact of different design concepts and alternatives on training requirements. This consideration is a key MANPRINT objective to assure reasonable manpower, personnel, and training demands of emerging training systems.

MANPRINT Techniques for Early Training Estimation

The goals and objectives of MANPRINT require that the manpower, personnel and training requirements of alternative weapon systems design concepts be accurately estimated. Early determination of training requirements, and their associated training resources, can help to optimize the design of the total weapon system. In addition to assessing the training impact of a design alternative, it is also important to identify which portions of the training requirement should be satisfied by embedded training. Our project is directed towards integrating active consideration of training into the earliest stages of the Life Cycle Systems Management Model (LCSMM) so that the design of the operational system and its supporting training system will be optimized.

In order to address the above objectives, we are developing a technique to provide an early estimate of the training requirement impact of a weapon system concept. This information will help us to assess the impact of this concept on Army training resources (cost, number of instructors, training devices, etc.). During the development of an Operational and Organizational (O&O) plan, or subsequently during the further refinement and development of weapon system design concepts, a framework is needed to consider the following aspects of MANPRINT for the early estimation of training requirements.

1. Number and type of MOS and/or quality level of personnel.

2. Jobs or tasks relative to the specificity of the emerging concept.

3. Identification/definition of man/machine interfaces.

4. Functions allocated to man or machine.

5. Knowledges and skills required by operator/maintainer functions.

The Early Training Requirements (ETR) data for a weapon system alternative will be stated at a gross level (i.e., functions and tasks) and be used for estimation purposes. The detailed training requirements (i.e., skills and knowledges required to perform the functions and tasks) would be detailed later as a part of training requirements

analysis. We are currently developing an "organizational framework" to identify what ETR data is required and to serve as a basis for integrating the ETR data. The completed database "framework" will represent the ETR from which: initial training resource estimations can be made, and early embedded training candidates identified. The above information will also be used to represent the necessary MANPRINT inputs for subsequent training requirements analysis and should be of form and character to support the simulation, if desired, of the man-machine interfaces represented.

Strategies for the Early Estimation of Required Training Resources

The purpose of this task is to develop techniques and tools which will use the ETR to identify initial training strategies so that estimates of required training resources can be made. These tools, currently under development, will allow designers to assess the impact of the ETR on individual training in the institution and the unit, and collective training in the unit. The tools are being designed to allow the training developer to define initial training strategies at a general/macro level. The macro training strategies will be developed relative to basic weapon system or functional classes. These estimates will be configured to permit rough relative training resource estimates to be made between competing weapon system candidates.

Early Estimation of Embedded Training Candidates

The purpose of this task is to develop a tool that will allow training developers to determine, in a timely fashion, the best candidates for embedded training. The tool will use as input the data associated with the ETR (described above). Issues in the development of this tool include:

- Within the range of training requirements identified for each weapon system class, which of the tasks and content domains are best suited for embedded training, taking into account the characteristics of the weapon system itself?

- To what level can these tasks be trained, taking into account the equipment characteristics, the environmental requirements, and the instructional needs of the trainees in both the active and reserve components?

TRAINING REQUIREMENTS AND TRAINING DEVICE STRATEGIES FOR WEAPON SYSTEMS

Effective conduct of the above MANPRINT activities and the selection of a weapon system candidate puts us in a position to define a training requirement and to conduct a training requirements analysis.

The proper statement of training requirements is critical for the effective conduct of a comprehensive training requirements analysis and the development of a training strategy. Therefore, we are developing standards and examples of how training requirements should be stated. We consider the following as important factors to be considered in the statement of a training requirement:

- Performance level to be achieved

- Environmental constraints

- Cost constraints

- Time to train

- Location of training (institution or unit)

We will develop operational definitions, with examples, to facilitate comprehensive training requirements analysis to support the development of training strategies and training subsystems.

Analysis of Training Requirements

The goals of this initiative are to: (1) collate and augment available techniques to assure the development of the prerequisite input data for the optimization models, (2) develop new techniques where required, and (3) develop a computer based system to guide an analyst (with appropriate examples) in developing the required input data for the optimization models.

We are aware that many techniques are available and we are attempting to identify techniques which can be used or adapted to provide the necessary input data to our models. We plan to develop computer models, with examples of well conducted training requirements analyses. Table 1 depicts the categories of data required by our optimization models and represents the expected outcome of the training requirements analysis effort. This analysis will be based on the functions and task database described below and shown in Table 2.

Functions and Tasks Database

This project will develop the databases necessary to support the conduct of a training requirements analysis. These databases will provide the necessary information to permit the development of the input information needed by the optimization models as shown in Table 1. The database is intended to supplement or complement existing data (e.g., MANPRINT). Where necessary, techniques will be constructed to help develop data not otherwise available. Examples of these data are shown in Table 2.

TABLE 1

INPUT REQUIREMENTS TO OPTIMIZATION MODELS

Task Requirements
- Tasks to be trained
- Performance criteria/standards
- Difficulty/criticality/frequency
- Safety considerations
- Skills and knowledges required

Trainee Characteristics Input

- Identified precursor skills and knowledges
- Entry level information (e.g., ASVAB)
- Training deficiency
- Aptitudes/learning rate
- Physical requirements

Training System Management Inputs

- Time allowed for training
- School resources
- Instructor requirements
- Number of students
- Cost constraints
- Available facilities
- Unit training considerations

TABLE 2

FUNCTIONS AND TASK DATABASE

- Missions grouped by functional categories

- Task listings by functional categories

- MOS used to operate/maintain end items (e.g., numbers, grade levels)

- Personnel statistics
  --demographics (age, sex, first language, etc.)
  --aptitude test scores (e.g. ASVAB)
  --time in service
  --physical capabilities

- Entry performance levels for tasks

- Standards and conditions for job performance

- Existing taxonomies and lists of skills and knowledge

- Data concerning existing training systems

Defining Training Strategy

Based on the results of a training requirements analysis, we are identifying the variables and considerations necessary for the development of a training strategy. We define training strategy as "a general comprehensive solution description to guide the development of training plans. The strategy gives consideration to a variety of factors such as policy, goals, constraints, resources and standards. It addresses the location, media, content, clustering, sequence and frequency of training." As a general solution, it would represent a top level training system design, and would serve as a higher level precursor of individual and collective training plans (ICTP). Our efforts in this regard would address individual training in the institution and unit. It will be designed to interface with the Computer-Aided ARTEP Production system (CAPS), which is a top down approach from collective training in the unit. The training strategy model would help the user to determine whether a training device or simulator is needed, and the appropriate role for embedded training.

The ICTP would represent the detailed and specific training plan developped by the appropriate Army School or Command. It would include detailed, specific information on tasks to train, training location, sustainment frequency, and appropriate supporting products, and would represent the training system design.

The training strategy tools to be developed should aid the decision maker and developer to:

- Identify all of the variables that affect an optimal training solution

- Comprehensively integrate training resources

- Adjust the training strategy over time to maximize effectiveness in response to shifts in constraints and criteria.

MODELS FOR THE OPTIMIZATION OF SIMULATION-BASED TRAINING SYSTEMS (OSBATS)

Whenever a training device or simulator may be required (based on a training strategy or school request) the OSBATS project is designed to answer the question: "How much is enough; do we need a $2,000,000 dollar simulator or is a $100,000 training device enough?"

Our OSBATS project has as its goals to: (1) optimize training systems by either achieving minimum cost for a desired performance level or maximum effectiveness for a given cost, (2) provide designers a flexible and comprehensive set of analytical tools with which they can interact, as needed during the design process, and (3) enable empirical and rational justification of a recommended approach. We have developed a set of models to support different aspects of the training device concept formulation process. We have oriented the model to the functions and processes that are important in any systems engineering effort to assure a cost/effective training device or simulator.

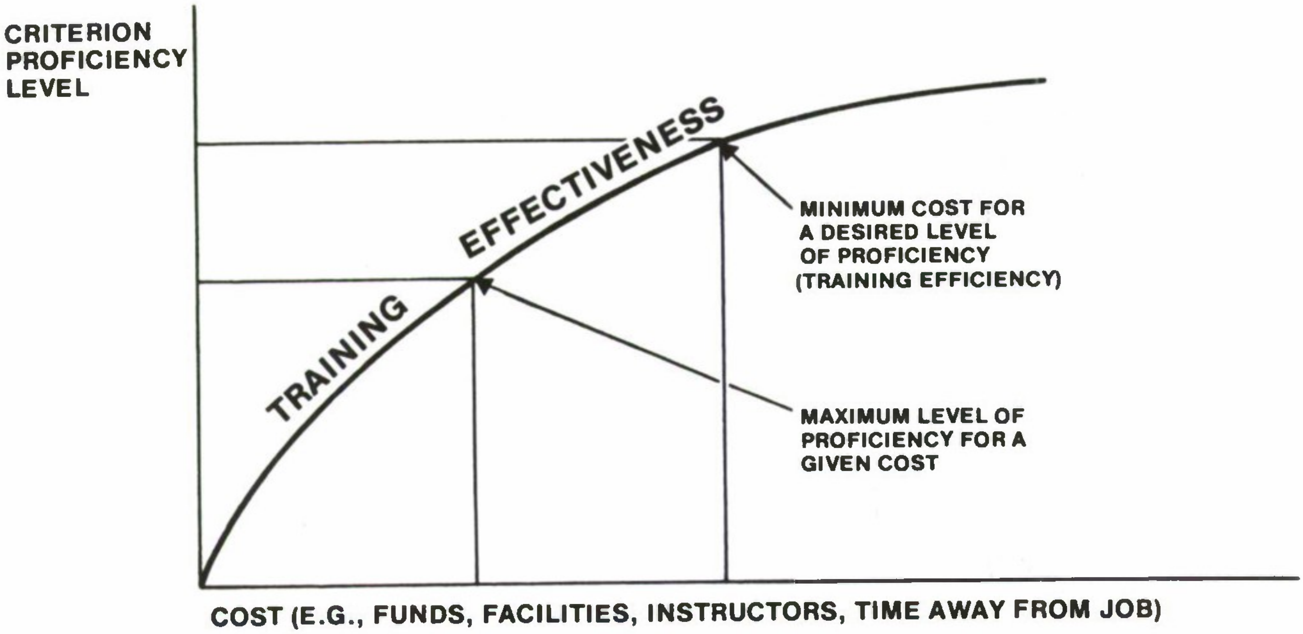


FIGURE 1. Cost-Effective Tradeoffs

Figure 1 illustrates the two issues central to the OSBATS model as well as to our overall program. What is the minimum cost for meeting a desired training goal? This is illustrated by point A. Secondly, what is the maximum training we can achieve at a given cost? This is illustrated by point B. We are seeking to develop the necessary data and techniques to permit the user to identify and evaluate design alternativews in an empirical manner.

At present, the OSBATS effort consists of five models. These are:

1. A Simulation Configuration Module that clusters tasks to be trained according to their need for training on a fullmission simulator (FMS), one or more part-mission simulators (PMS), or actual equipment (AE).

2. An Instructional Feature Module that determines the relative priority with which instructional features should be included in a training device.

3. A Fidelity Optimization Module that determines the relative priority of features that allow a training device to represent aspects of the operational environment.

4. A Training Device Selection Module that selects the training devices that can be used to meet the training requirements for each task at the least cost.

5. A Resource Allocation Module that determines the optimal allocation of training time to training devices and actual equipment to meet all training requirements, considering constraints on device procurement and use.

These models are currently in prototype form and are being evaluated by PM TRADE engineers. Their application is currently limited to the aviation domain. We plan to start developing first generation models and expanded data bases for use by PM TRADE, TRADOC and Army schools in the following year.

The concept of operation for the OSBATS model is based on iterative use of the five modeling tools. This iterative concept reflects the interactions inherent in the training-system design process. The tools may be used in a variety of orders, tools may be used several times, and inappropriate or unwanted tools may be bypassed. When used by different analysts who make different assumptions, the OSBATS model will produce different recommendations. However, the model will document the assumptions and rationale that underlie each recommendation.

As the name denotes, OSBATS focuses on simulation-based training systems.* The underlying models and the OSBATS output emphasize training system characteristics for representing the task or mission, in the system and environment that the student must eventually use. The representation is an analog of the real task/mission, system, and environment. We plan to expand OSBATS into the arena of training devices that do not necessarily employ simulation. Many of these devices tutor, instruct, or present new material to the student with little or no representation of real-world tasks,

*Personal communication with P. Sticha, 20 August 1987.

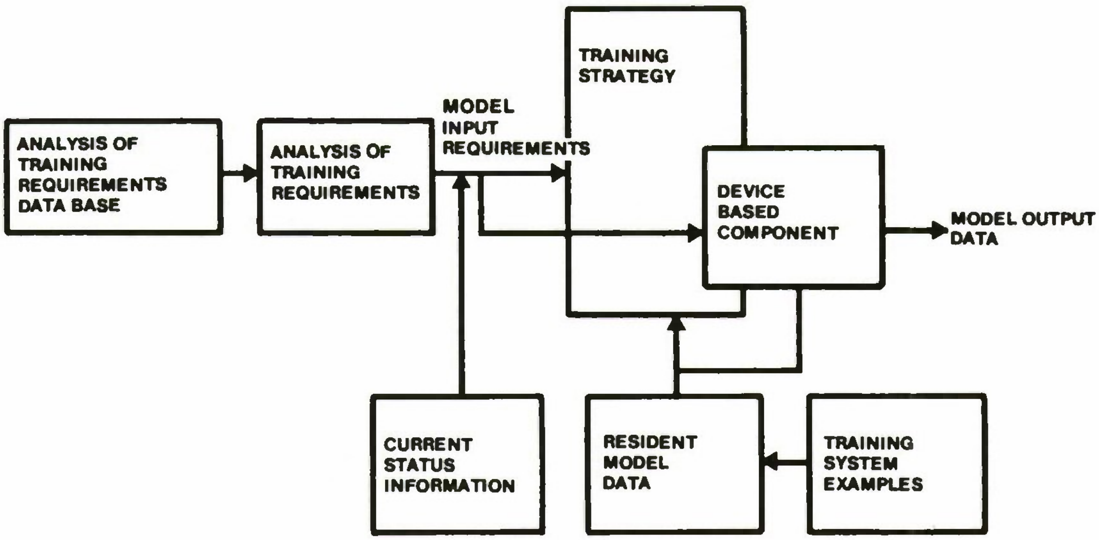


FIGURE 2. Training System Optimization Models and Data Bases

missions, systems, or environments. A model is required to cluster tasks according to their skill requirements, thus identifying common skills that may be addressed by a training device. The following additional models are planned to represent this capability as well as other desired capabilities:

1. Skill-Based Task Organizing Module
2. Skill-Based Training-Device Design Module
3. Instructor/Operator Station Design Module
4. Embedded Training Evaluation Module
5. Embedded Training Design Module
6. Rough-Order-of-Magnitude Cost Estimation Module

DATA BASE DEVELOPMENT FOR THE OPTIMIZATION OF TRAINING SUBSYSTEMS AND TRAINING DEVICES

The developmental efforts described above (Models for Training Strategy Development, and Models for the Optimization of Simulation Based Training Systems) require data bases from which the necessary information for optimization can be drawn. The goals of the required database efforts are to detail the internal data and rules required for the models; to identify or develop methods for collecting, converting or transforming the data; and to define the necessary framework for organizing the varied rules and data required by the different optimization efforts. The discussion below speaks directly to the databases required for the OSBATS. The Training Strategy Development model will also require such databases. However, this project is still in its formative stages.

Figure 2 shows the functional relationship between our planned models and database requirements.

There are two types of data required to support the functioning of the the OSBATS models. The first type includes the basic data involved in the generation of options, tradeoffs, and configurations. These are general task characteristics, types of fidelity options and associated costs, learning parameters, types of instructionsl features and their costs, etc. These data will come from thorough analytical evaluations and data collection efforts, including experiments designed to verify certain parameters (e.g., learning parameters) and the hypothesized relationships within the model. The second includes aggregated relationships between the databases in the form of rules. These rules are expected to be expert system type production rules. They will be structured for use in specifying types of instructional features, fidelity levels, etc., for different tasks and task groupings. The aggregating rules, resident data, and input data are interconnected in several ways. The rules are used for relating input data about tasks to resident variables about training device features and costs, as well as combining input data and resident information in order to generate process data for the analysis session.

The data base management system (DBMS) will support access through the models for user inspection and entry into the data bases. The system will support data tracking, validity checks, and other typical user-friendly features. This interface will include the delivery of data to the models upon demand from the models (for example, in the form of program calls), the acceptance by the DBMS of generated or input data from the models, and the provision of inspection and editing facilities through the model interface. The aggregate rules will also be inspectable through the models, and editable by the existing OSBATS expert systems shell mechanisms.

EMBEDDED TRAINING

Current Army policy advocates the consideration of embedded training as a first alternative but not its exclusive use. Policy further requires that embedded training: (1) will not interfere with the operational requirements/capabilities of the system, and (2) will train individual tasks through force level tasks as required. The policy letter defines embedded training as "Training that is provided by capabilities designed to be built into or added onto operational systems to enhance or maintain the skill proficiency necessary to operate and maintain that equipment end item."

Moreover, Army policy sets these goals:

- Include a training strategy in the O&O plan and develop training requirements and resources during system concept formulation.

- Analyze and provide a rationale for either including or not including embedded training at each materiel decision process milestone.

- Identify the MANPRINT and Integrated Logistics Support (ILS) processes as the catalyst for considering embedded training in the pre-concept formulation and subsequent prototyping phases.

PM TRADE has established the following objectives relative to embedded training, which is consistent with our overall program objectives described above:

- Achieve a top-down systems engineering approach to the definition and development of <u>training systems</u> at all levels beginning in earliest concept phases.

- Integrate training strategies and resources across functional areas to avoid redundant capabilities.

- Ensure the fullest integration of available technology, device, simulators and embedded training to achieve the most effective training at the lowest cost.

Our research objectives are to: (1) identify under what conditions embedded training should, or should not be included in weapon systems under development, (2) identify functions and tasks (by weapon system class) which best lend themselves to embedded training, (3) identify critical design tradeoffs related to embedded training, and (4) organize current and existing information relating to embedded training.

Our joint project in embedded training is designed to provide insight into such questions as:

How does embedded training best complement other training techniques (i.e., the best mix of training programs, devices and embedded training)?

What are the implications of different engineering configurations for classes of materiel end items (e.g., Armor, Artillery, Aviation, Communications)?

What are the optimum formats for presenting embedded training information (e.g., job aids vs. training)?

What are the interface requirements among materiel, combat and training developers?

What are the special implications of categories of embedded training such as individual, team functional and force level?

What are the engineering, operations, and logistics impacts in terms of life cycle costs, reliability and supportability?

To date our joint program has taken a major first step. Several reports have been prepared which address: (1) The development of embedded training in exemplar systems such as Fiber Optic Guided Missle (FOG-M), (2) laboratory technological research and surveys to establish the state-of-the-art in emebedded training, and (3) the development of specifications for the inclusion of embedded training in weapon systems. Representative reports include Findley, Alderman, Bolin, and Peckham (1985), Carroll, Harris, and Roth (1986), Massey, Harris, Downes-Martin and Kurkland (1986), and Purifoy, Harris, and Ditzian (1986).

ORGANIZATION OF TECHNICAL INFORMATION FOR ELECTRONIC DELIVERY

Army policy will soon require that information now contained in paper technical manuals be delivered electronically. Both technicians and logistics specialists have long been unsatisfied with technical manuals because they difficult to use, hard to update, and bulky. On the other hand they can present such information as flow diagrams and large schematic drawings far better than any other medium. Weapon systems entering the Army's inventory in the 1990's will be maintained by technicians using computers to access their technical information rather than technical manuals. Kincaid and Braby (1987) discussed a number of ramifications of this policy including user acceptance issues, methods of automating authoring and delivery of the technical information, and techniques for organizing and presenting the technical information. These issues apply equally to information now contained in technical manuals and embedded training. For example, job performance aids are now routinely presented in technical manuals (e.g., in New Look manuals) and are also considered an appropriate format for embedded training.

The primary technical objectives of a project we are just getting underway are to: (1) create and demonstrate techniques for the optimal organization of technical information, (2) create computer-based job performance aid display algorithms appropriate for, and compatible with, electronic devices for delivering technical information, and (3) demonstrate these information organization techniques and display algorithms for job aids, including embedded training.

To achieve these objectives we are: (1) assembling examples of types of technical information (e.g., fault isolation, repairs, operation, casualty procedures, installation, scheduled maintenance) and analyzing and describing the process for presenting this TI using a microcomputer; and (2) describing and illustrating efficient ways to organize and present this information to the user.

This project supports three PM TRADE initiatives for the delivery of technical information: (1) MEIDS - the militarized/ miniturized information delivery system, (2) EIDS - the Electronic Information Delivery System, and (3) embedded training. The concept for MEIDS is currently being formulated; however, it is envisioned as a portable microcomputer to replace paper technical manuals for maintenance of equipment in the field. EIDS devices are videodiscs controlled by microcomputers, supported by courseware development software.

SUMMARY

We have described a series of inter-related projects the purpose of which is to optimize the process for conceptualizing, designing and procuring training systems, including the consideration of training strategy. We are developing computerized decision aids for the training developer, as well as the data bases which are needed to operate the decision aids.

Each of our major projects are being related to each other, with respect to the following considerations: (1) system relationships, (2) data input and output requirements, (3) processing and procedural requirements, and (4) database requirements.

Early training estimates, training requirements analysis, including embedded training will have common database requirements, and somewhat similar analytical procedures. They differ in the level of detail and generalization relative to their stage of use. OSBATS, while an intergal part of this system, will have different data requirements and analytical procedures. However, it depends upon the outputs of the other processes for its initiation.

We are relating our research and development efforts to the weapon and training system development procedures of the Army, that they are designed to support. This includes the Required Operational Capability (ROC), Organizations and Operations (O&O) plan, Training Device Needs (TDN) and Individual and Collective Training Plan (ICTP). The relationship between the technology base and operational base will be made explicit at every point.

PM TRADE and ARI, with the active support of the Army training community, are collectively applying a systems engineering focus to the training development process.

REFERENCES

Andrews, D.H., Singer, M.J., Ozkaptan, H., and Hofer, R.C. (1987) "Optimization of Training Systems and Subsystems: Models, Methods and Data Bases." Paper presented at the International Conference on Military Personnel and Training, Luxembourg, and published in the Proceedings.

Braby, R., Henry, J.M., Parrish, W.F., and Swope, W.M. (1975) A Technique for Choosing Cost-Effective Instructional Delivery Systems. TEAG Report No. 16. Orlando, FL: Navy Training Analysis and Evaluation Group.

Carroll, R.C., Harris, C.B., and Roth, J.T. (1986) Draft Functional Specification and Data Item Description for FOG-M Embedded Training Subsystem. Research Product. Alexandria, Virginia: Army Research Institute.

Finley, D.L., Alderman, I.N., Bolin, S.F., and Peckham, D.S. (1985) Embedded training and systems acquisition. Army R,D,&A.

Gagne, R.M, and Briggs, L.J. (1979) Principles of Instructional Design. (2nd Ed.) New York: Holt, Rinehart and Winston.

Kincaid, J.P., and Braby, R. "Computer-based authoring and delivery of technical information." (1987) Paper presented at the International Conference on Military Personnel and Training, Luxembourg and published in Proceedings.

Massey, D., Harris, M., Downes-Martin, S., Kurland, L. (1986) Embedded Training Technology Survey. Working Paper. Alexandria, Virginia: Army Research Institute.

Purifoy, G.E., Jr., Harris, G.B., and Ditzian, J.L. (1986) Design Concepts for FOG-M System Embedded Training. Working Paper. Alexandria, Virginia: Army Research Institute.

ABOUT THE AUTHORS

Dr. Ronald C. Hofer is Chief of the Operations, Research and Engineering Management Division for the Army Project Manager for Training Devices. Currently, one of his primary interests lies in developing an integrated, cohesive approach for acquiring training devices, simulators and embedded training within a total training system context. In addition to his extensive experience in simulator and training device technology and engineering developments within the Army and other services, he has held senior engineering positions within the Office of Project Manager for Army Tactical Data Systems and the Army Electronics Technology and Devices Laboratory. Dr. Hofer holds the BSEE, MSEE and Ph.D. degrees from Purdue University, West Lafayette, Indiana.

Dr. Halim Ozkaptan is Chief of the Army Research Institute Field Unit that is collocated with PM TRADE in Orlando, FL. He has had extensive experience as the manager of R&D programs for the Army and Navy. He was formerly Chief of Human Factors at Martin Marietta and a program manager for Grumman Aircraft and Republic Aviation. He has been a lecturer at Maryland and Catholic Universities, an invited NATO lecturer, and has over 60 professional publications and presentations. He has represented DoD on International Data Exchange Agreements. He has a Ph.D. in Applied Experimental Psychology from Catholic University.

Dr. J. Peter Kincaid is a research psychologist with the Army Research Institute, Orlando Field Unit. He received his Ph.D. in Human Factors psychology from Ohio State University and has also worked for the Navy and the Air Force as a researcher in training and human factors. He has numerous professional publications and serves as an adjunct faculty member at the University of Central Florida.

HYBRID ADA/FORTRAN SYSTEMS FOR FLIGHT SIMULATION

Vincent F. Rich
Gould Inc., Information Systems
Computer Systems Division
6901 West Sunrise Boulevard
Fort Lauderdale, Florida 33313-4499

ABSTRACT

Contemporary military flight simulators are normally programmed in the FORTRAN language. The Ada® language has been mandated by the Department of Defense and is expected to be in widespread use by 1990.

Ada supports an interface to subprograms written in other languages. This multilingual capability will allow simulator vendors to phase the conversion to Ada over a number of projects, providing that a hybrid system is acceptable to the end user. This capability will also allow simulator upgrades to be programmed in Ada while the existing software remains largely unchanged.

The phased conversion of a simulator from FORTRAN to Ada can be accomplished with either a lateral, top down, or bottom up strategy. The lateral strategy involves the structuring of the software into a number of operating system processes communicating via shared memory. These processes can then be programmed in either Ada or FORTRAN. The top down strategy involves high level Ada programs calling lower level FORTRAN subprograms such as standard software components and math models. The bottom up strategy involves the conversion of the standard software components into Ada, and the calling of these components from a high level FORTRAN program. Selection of the optimum strategy will depend on a number of factors including the computer system architecture, operating system, and characteristics of the FORTRAN and Ada compilers.

The advantages of a hybrid system must be balanced against the possible loss of reliability and maintainability of the software. Potential problems exist in the areas of exception processing, parameter passing, constraint checking, FORTRAN/Ada runtime system conflicts, and concurrency.

This paper considers the advantages and disadvantages of each implementation strategy and discusses the problems and difficulties that are encountered in the implementation of a multilingual system.

INTRODUCTION

Most military flight simulators are programmed in the FORTRAN language and employ small amounts of assembly language for low level functions.

Typically, the real-time software is structured as a collection of subroutines. These subroutines are grouped into a number of operating system processes, and within each process the subroutines are sequenced in a predetermined order by an executive routine. The subroutines communicate with each other through common data regions. High fidelity flight simulation requires rapid and deterministic execution of the software. The design of the software and the coding standards employed are strongly influenced by performance criteria.

ADA

Ada is a standard (1) high order programing language mandated for use in embedded computer systems by the Department of Defense. Ada based proof of concept simulators are currently under development, and Ada is expected to be in widespread use by 1990.

Ada supports features which encourage and enforce good software engineering practices. The increase in expressive power and the richness of the language allow Ada programs to be both smaller and easier to comprehend than equivalent FORTRAN programs. The improved software engineering methods and easier program comprehension are expected to yield significant cost savings over the total life of the system since software maintenance costs contribute substantially to the cost of ownership of the system.

®Ada is a registered trademark of the U.S. Government (Ada Joint Program Office)

Unfortunately, current implementations of complex Ada language features such as inter-task synchronization (a 'rendezvous') may be too slow for stringent real-time applications such as flight simulation. Similar capabilities can be provided by calling small fragments of programs written in other languages from Ada. For example, as an alternative to a rendezvous, task synchronization can be achieved through the use of assembly language routines implementing hardware semaphores (2).

MIXED LANGUAGE PROGRAMMING IN FLIGHT SIMULATION APPLICATIONS

The Ada standard allows Ada programs to call program modules written in other languages (3). The range of languages supported and limitations imposed upon the user are implementation dependent and vary from compiler to compiler. For flight simulation purposes, the ability to call assembly language and FORTRAN is desirable.

The ability to call assembly language routines allows the direct use of machine instructions such as input/output device commands and semaphores. An assembly language interface can also be used to call operating system services in an implementation where this facility is not directly supported by the Ada compiler.

The ability to call FORTRAN allows the reuse of existing code.
This provides significant economic and logistical advantages to the implementer of the system, but dilutes the long term economic advantages of using Ada.

Some future flight simulators will probably use an implementation of the Unix™ operating system with real-time extensions. The ability to interface Ada programs to routines written in the "C" language will be useful, as it will allow the flight simulation code to interface with the Unix system services through the Unix "C" libraries.

ECONOMIC AND PROJECT MANAGEMENT ISSUES

Simulator vendors frequently reuse generic software components and utility programs that have been used in previous projects of a similar nature. These components and programs are mature and are unlikely to require future maintenance. Some of the components may use cunning techniques and algorithms that were specifically designed to increase execution speed in a FORTRAN environment and are not readily convertible into Ada.

Other components such as motion system software may require extensive retesting in order to verify correctness after they have been rewritten in Ada. The ability to mix Ada with FORTRAN will allow a phased transition from FORTRAN into Ada and this in turn will allow the cost and risk of converting the standard software to be amortized over a number of projects.

Note, however, that the purchaser of the system must balance the savings in initial purchase cost and possibly more attractive delivery schedule against the reduction in total life cycle cost savings caused by the hybrid implementation. Additional costs will be incurred in the provision of a dual programming environment, and the maintenance staff will need to be proficient in both languages.

The ability to mix Ada with FORTRAN will allow simulators to be incrementally converted into Ada as new software modules are developed. This is particularly useful in engineering and research simulators where Ada allows rapid prototyping of new software modules that can then be easily integrated into the existing software structure.

This capability can be used to test Ada software for avionic applications on non-Ada based engineering simulators. The Ada program can initially be compiled to run on the simulation computer and can interact with the rest of the simulator written in another language. Once the basic testing is complete, the Ada software can then be recompiled to target the processor within the black box and the testing can be continued with the black box attached to the simulator.

In-service simulators can be field upgraded with new subsystems written in Ada while minimizing the cost of the upgrade and the out-of-service time.

Software packages produced by parties other than the simulator or computer vendor may be used in the simulator, for example, to interface to graphics terminals. A mixed language programing environment allows the continued use of these packages prior to their conversion into Ada.

Non-Ada software components are not subject to the strict compilation order rules that are applied to Ada components. In the Ada case, a program unit must be recompiled if the specifications of any referenced program units are recompiled.

As all Ada program units have an implied dependency upon the runtime (through package STANDARD) the provision of a new version of the runtime by the computer vendor can force the recompilation of all of the Ada code. If a third

™Unix is a trademark of AT&T Bell Laboratories

party package is written in Ada, then the vendor of that package will be required to supply a binary package compatible with the new runtime. Alternatively, the package sources so that the simulator vendor can recompile the package. If the third party package is written in a language other than Ada, the compilation order rules are not enforced and revised binaries or package sources are not required.

METHODS OF INTERFACING ADA AND FORTRAN PROGRAMS

Three strategies can be used to interface Ada and FORTRAN programs. The simplest strategy calls for the division of the Ada and FORTRAN programs into separate operating system processes. Various methods of interprocess communication are possible, depending on the nature of the programs and the computer system architecture. A more complex strategy is to mix Ada and FORTRAN programs within one operating system process and allow the FORTRAN programs to be called by the Ada programs. The third strategy is to mix Ada and FORTRAN programs within one operating system process and to allow the Ada programs to be called by the FORTRAN programs. Each of these methods is now examined in detail.

SEPARATE OPERATING SYSTEM PROCESSES

Process Interfacing Through Disc Files. Figure 1 shows a non-Ada utility process communicating with a real-time process written in Ada. A typical utility process (such as a Ground Station Data compiler) processes source text and stores a binary representation into a disc file. The contents of this file can then be accessed by the real-time program as required. The file contains an array of data structures possibly of different types and lengths. The real-time process must be written so as to take the existing data structures and convert them into equivalent Ada structures. The programmer can create a set of Ada records to match the content of the disc file but needs to understand how the Ada compiler actually forms the records (in terms of bounding, packing, etc.) and also be alert for variations in the representation of different types of data. The programmer must verify that the length and internal format of each data item is consistent between the Ada implementation and the data file.

Some programming effort and the requirement to understand the internal structure of an Ada record can be alleviated if the Ada compiler supports record representation clauses. This implementation dependent feature of the language allows the explicit definition of an Ada record.

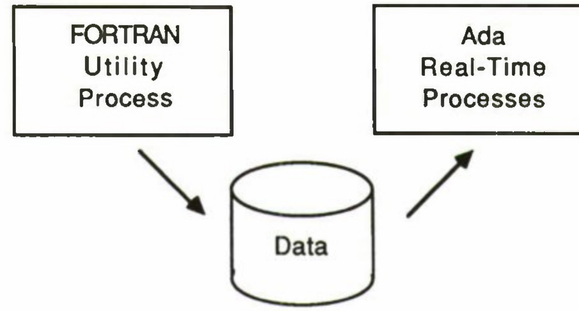


Figure 1.
Mixed Language Interface via Disc Files

The Ada language supports comprehensive run time checking. If a variable exceeds its predefined range, an 'exception' is raised and normal processing is terminated. The exception is processed by an exception handler designated for the particular zone of the program where the exception occurred. A range check is normally applied after the new value of the variable has been computed. If data is imported from an external source, then no range checking has been performed prior to the use of a data item and out of range values can lead to unexpected program failures that cannot be easily associated with the particular out of range variable. To avoid this type of program pathology, the incoming data items should be explicitly checked for correctness.

Interfacing Processes on a Loosely Coupled System. Figure 2 shows a computer system executing real-time Ada programs communicating with a system executing programs written in another language. A realization of this configuration is a flight simulator computer transmitting packets of data to a computer generated image type of visual system. The primary difference between this method and the preceding method is that the data packets are transferred by direct calls to the operating system rather than calls to the standard Ada input/output packages. The same considerations about record structure apply, as an Ada record corresponding to the packet

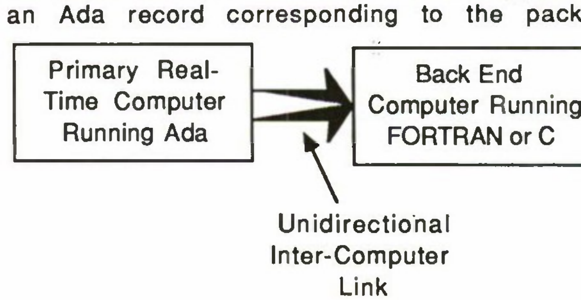


Figure 2.
Mixed Language Interface with Loosely Coupled Computers

format expected by the non-Ada system needs to be constructed.

Interfacing Processes on a Tightly Coupled System. Figure 3 shows multiple processes executing on the same computer system and communicating via shared memory. Some processes may be written in Ada and some in FORTRAN. The Ada processes access the shared region through an Ada record that has been structured to exactly match the layout of the global common used by the FORTRAN programs. Using pointers or address representation clauses, the record is positioned to the same logical address space as the FORTRAN global common.

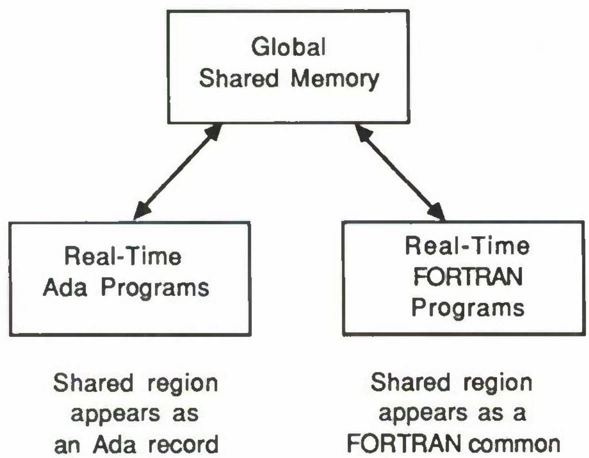


Figure 3.
Mixed Language Interface
with Shared Memory

CALLING FORTRAN FROM ADA CODE WITHIN THE SAME PROCESS

An Ada program may call a FORTRAN program by using the Pragma Interface feature of the language. This feature is optional and the degree of the support varies from implementation to implementation. The connection between the Ada and FORTRAN programs is established by use of Pragma Interface. An Ada procedure consists of a specification and a body. In order to call FORTRAN from Ada, an Ada specification is prepared for the FORTRAN routine. This specification describes the name of the routine as called by the Ada program, the actual name of the FORTRAN routine, and the number, type, and order of the parameters to be passed. A sample specification is shown in figure 4. The Pragma Interface statement specifies that the procedure specification does not have a related body and that the compiler should insert a call to the FORTRAN module wherever the Ada name for the FORTRAN routine is used. The compiler

```
-- This is the Ada specification for a max
-- rate FORTRAN engines module that
-- receives the engine number as an input
-- parameter.
--
Procedure Engines_1R1(Engine: Engine_Number);
Pragma Interface (FORTRAN, Engines_1R1);
--
-----------------------------------------
-- In the main executive, the FORTRAN
-- engine module is called for engine No. 3
-- by the statement:
--
Engines_1R1(3);
```

Figure 4.
Example of the use of
Pragma Interface

inserts instructions at the call to provide the correct interface between the two languages. The compiler uses the language keyword to select the appropriate instructions. This interfacing software resolves some of the differences that exist between the underlying implementations of the two languages. For example, the Ada compiler may generate code that conforms to certain conventions regarding stack size and direction of growth. The FORTRAN compiler may employ different conventions, and so a correct FORTRAN execution environment must be established at the moment of calling the FORTRAN routine and the Ada environment regained when a return is made to the Ada procedure. The programmer is responsible for resolving any incompatibilities in parameter passing that the compiler cannot handle, for example, differences in the orientation of arrays.

Using the Pragma Interface capability, a single process can be constructed from a mixture of Ada and FORTRAN. Figure 5 shows the simulator specific upper layers of the software written in Ada, calling the generic software components written in FORTRAN.

The Ada and FORTRAN code segments can communicate with each other via Ada records and FORTRAN global commons that are mapped to the same memory locations. Alternatively, the Ada code may pass parameters to the FORTRAN routines, but this may be less efficient, and also may require changes to the FORTRAN code.

Some problems may be encountered with the handling of program exceptions. An executing FORTRAN routine called from an Ada procedure may encounter an error and cause a machine trap. These may range in severity from an arithmetic exception to the attempted reference of non-present memory or the execution of a privileged instruction. The method of processing

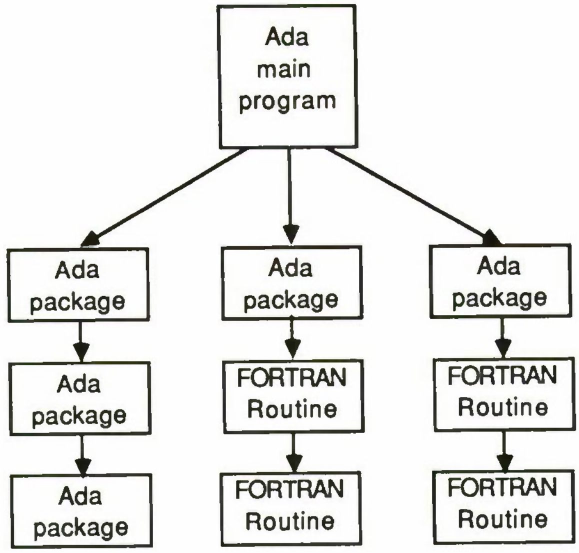


Figure 5.
Top Down Strategy

these traps varies from implementation to implementation. In the crudest implementation, a trap will have undefined results and possibly cause an abort of the whole program. A reasonable implementation is one in which at least some of the traps are converted into the equivalent Ada exceptions and then propagated through the Ada program according the rules of the Ada language. The exception should be made to appear to have occurred at the point in the Ada procedure where the FORTRAN routine is called. If the program under test does not contain any exception handlers, the runtime system should report the name of the FORTRAN subroutine that caused the exception, the FORTRAN subroutine call tree that led to the failing subroutine, and the Ada procedure call tree that lead to the first subroutine in the FORTRAN call tree.

Some existing FORTRAN routines may actually generate arithmetic exceptions during normal operation. In a FORTRAN environment, such exceptions are often ignored and the result of the computation clamped to the minimum or maximum value for the particular variable being computed. Execution of the same FORTRAN code in an Ada environment will cause termination of the FORTRAN code and an exception to be raised. Suppressing runtime checks with the Ada Pragma Suppress may not allow normal execution of the FORTRAN code as the pragma may only suppress the insertion of additional machine instructions to perform runtime checks. The pragma may not suppress the processors internal arithmetic exception and trapping logic and so traps will continue to occur when arithmetic operations generate exceptions. The suggested method for resolving this type of problem is to modify the FORTRAN code so as to perform checks on the input variables of any operation which has the potential to cause a valid arithmetic exception. Note that the provision of an Ada exception handler with a null body will not resolve the problem, as the FORTRAN code following the site of the exception will not be executed.

A further area worthy of investigation is the handling of concurrency and reentrancy by the FORTRAN programs. Ada supports the concept of tasks which can be executed concurrently in a multiprocessor system or in an interleaved manner on a uniprocessor system. The FORTRAN language does not support either concurrency or reentrancy. Ada tasks can call the same FORTRAN routine and this raises the possibility of program malfunction due to the corruption of static memory locations allocated for use by the routine. Fortunately, most contemporary implementations of the Ada tasking model only allow a task switch when a scheduling event occurs such as a rendezvous, task abort, or initiation of a delay. Sections of code between scheduling events are protected as a task switch cannot occur and therefore any FORTRAN code called by Ada code cannot be interrupted and reentered.

Unfortunately, advanced implementations of the Ada tasking model support full asynchronous operation and true concurrency in a multiprocessor environment. If a FORTRAN routine is to be shared between two Ada tasks then the programmer must investigate the internal implementation of the FORTRAN in order to determine if support for reentrancy is provided. For example, if a particular implementation stores a routines return address in a dedicated memory location, then a task switch and second call to the routine will destroy the return address established by the first task. If reentrancy cannot be confirmed, then the FORTRAN routines should be replicated and each Ada task provided with its own set.

The last technical issue that needs to be considered is the conflict that can exist between the Ada and the FORTRAN runtime systems. The basic structure of a process written in a high order language is shown in figure 6. The application code generated by the user is linked to a copy of the runtime system. The runtime system converts user requests into operating system service calls that perform the desired function. Whenever possible, computer vendors attempt to build runtime systems that can be used by multiple languages. However, the requirements of Ada are so unique that most vendors offer a separate runtime for Ada.

The intermixing of Ada with FORTRAN can

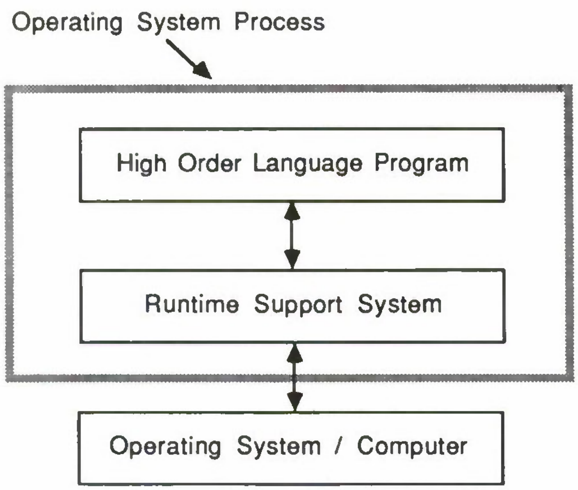


Figure 6.
Structure of a
High Order Language Process

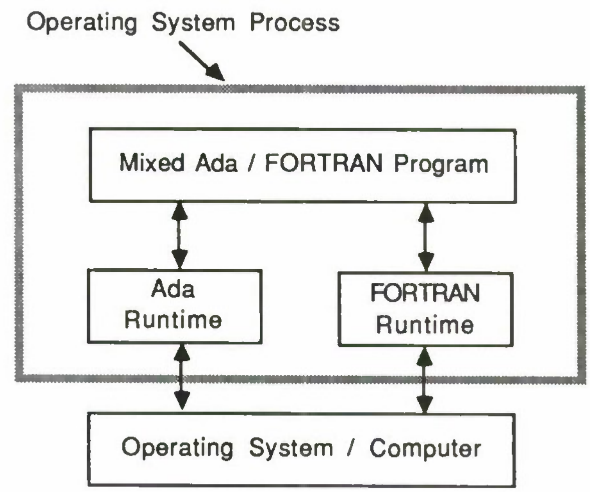


Figure 7.
Process with Dual
Runtimes

lead to a situation where one process contains copies of both the Ada and FORTRAN runtime systems. This can occur when an Ada program calls a FORTRAN routine and that routine uses a language feature that calls the FORTRAN runtime system. The linking process calls in both the Ada and FORTRAN runtime systems and the resultant program structure is as shown in figure 7. During the execution of the process, both runtime systems can allocate system resources, without regards to the needs of the other runtime. For example, both runtime systems can attempt to write to the same peripheral device. If either runtime maintains data buffers or device status information, then that data can become invalid when the other runtime interacts with the device. The first runtime will then malfunction when it attempts to interact with the device using obsolete data. These conflicts can be avoided by limiting the FORTRAN code to those language features which do not call any operating system services. This limitation is in harmony with FORTRAN code for flight simulators as the bulk of the code is performing arithmetic and logical operations on data items held in common memory. Wherever possible, input/output operations should be limited to the Ada portions of the program. If this is not possible (for example, in the case of a FORTRAN graphical support package for a terminal), then the Ada program should not attempt to access devices that are used by the FORTRAN portion of the program.

If the target computer system is operating in bare machine mode (that is with no operating system at all) then all input/output must be performed by the Ada program as the FORTRAN runtime can no longer call the operating system. Any attempts at calling the nonexistent operating system should be converted into Ada exceptions and passed back into the program for recovery.

CALLING ADA FROM FORTRAN CODE WITHIN THE SAME PROCESS

Some computer systems support the ability to call an Ada procedure from a FORTRAN routine. This allows a simulator to be converted into Ada from the bottom upwards, by starting the conversion with the low level routines. While such a scheme is technically feasible, it does not offer the economic advantages of the reuse of standard FORTRAN code. The non-support of Ada language features such as tasking and exception handling by FORTRAN programs, severely limit the utility of a bottom up strategy.

OPTIMIZATION ISSUES

The programmer needs to be alert for failures that can be introduced by improving the execution speed of an Ada program with a code optimizer. The optimizer, which has no knowledge of the FORTRAN program, may decide through data flow analysis, that a particular variable is not accessed elsewhere in the Ada program and therefore need not be computed. This data item will not be stored in the memory region shared between the Ada and FORTRAN programs and the FORTRAN program will malfunction. The reverse situation can occur when an optimizer is used on the FORTRAN program.

CONCLUSIONS

The ability to mix Ada and FORTRAN code will allow a phased transition from FORTRAN to an all Ada flight simulator. This will reduce the cost of initial Ada implementations while accelerating the introduction of Ada in flight simulation.

Communicating processes written in different languages provide an elegant method of implementing a mixed language system but the Ada and FORTRAN components have to be divided into fairly large segments.

Use of the Ada Pragma Interface allows both Ada and FORTRAN code to be mixed in the same process. Awareness of problems in the areas of exception processing, concurrency, and runtime system conflicts, is required.

Both the communicating processes approach and a top down approach can be combined. This allows the use of large software components (for example, motion system software) to be used as an independent process while another process composed largely of Ada can call support routines written in FORTRAN.

A bottom up strategy can be employed but offers no advantages over a top down approach and does exhibit technical difficulties.

BIBLIOGRAPHY

(1) ANSI/MIL-STD-1815A, Ada Programming Language.
(2) R.C. Holt, G.S. Graham, E.D. Lazowska, M.A. Scott, "Structured Concurrent Programming", Addison-Wesley, Reading, Massachusetts, 1978, pp. 26-30.
(3) ANSI/MIL-STD-1815A, Section 13.9.

ABOUT THE AUTHOR

Mr. Rich is Manager, Ada Compiler Technologies, at Gould Inc. He holds a M.S. in Computer Science and an M.B.A. from Nova University. He has twenty years of software and systems engineering experience including compilers, operating systems, and real-time software for commercial and military flight simulators. Since joining Gould in 1981, he has been involved in the development of the MPX-32 operating system and several Ada related projects. He is currently heading a group responsible for the development of Ada compilers, runtime systems, and related software.

REUSING FORTRAN IN AN ADA DESIGN

William F. Parrish, Jr.
Naval Training Systems Center
Orlando, FL 32813-7100

Kent G. Trewick
Naval Training Systems Center
Orlando, FL 32813-7100

ABSTRACT

Many simulator processes and algorithms have been implemented in FORTRAN. Some examples are ocean models, aircraft avionics models, and sonar sensor models. As we begin writing training device software in Ada[R], it is important that we consider reusing existing FORTRAN code. This is particularly true for FORTRAN based trainers undergoing major software modifications. Various techniques for interfacing Ada and FORTRAN designs are investigated. Benchmarks are presented comparing an all FORTRAN or all Ada implementation to a combined FORTRAN/Ada implementation. Problems concerned with calling FORTRAN subroutines from Ada procedures and tasks and vice versa are explored. Differences in arithmetic types between the two languages are also explored. Particular emphasis is placed on the effect that a combined Ada/FORTRAN implementation has on computer resources. This consideration is of major importance when modifying an existing trainer where spare time and memory may be very limited.

INTRODUCTION

Current Navy and DOD instructions require that weapon system training device software be written in Ada. In some instances the reuse of existing FORTRAN code should be considered. There are primarily two cases where this requires consideration. The first case is that of a major software modification where most of the existing software is written in FORTRAN. OPNAVINST 5200.28 requires that Ada be used for major upgrades. A major upgrade is defined by DOD Directive 3405.2 as a redesign or addition of one-third or more of the software. The other case where the reuse of FORTRAN is important is in those instances where it is desirable to use existing FORTRAN models. Many potentially reusable FORTRAN models exist. For example, the Naval Training Systems Center has an ocean model that is written in FORTRAN that has been furnished as Government Furnished Information on several training device contracts.

Numerous factors should be considered when integrating FORTRAN and Ada designs and code. This paper examines many of these factors. A series of benchmarks were run and results are compared. Memory requirements and execution times are compared between languages.

BENCHMARK DESCRIPTIONS AND RESULTS

Three benchmark programs were used. The first benchmark program is a simple program that computes the number of prime numbers in a specified range. This program was chosen because of simplicity and the ease with which execution time can be varied. The prime number program also facilitates easy comparison of arithmetic types. The second benchmark program is a mathematical routine that calculates a simple sine and exponential function using infinite series. This program was chosen because it represents a cyclic activity when used to generate a table of values. The third benchmark is a modified version of the Dhrystone (9) benchmark. Modifications were made to the benchmark by the University of Central Florida to closely approximate the mix of high level language statements found in a typical training simulator program.

The first two benchmarks were implemented with and without Ada tasks. This was done so that a comparison could be made between the execution times with tasks on a parallel processor machine and a single processor machine.

Several different implementations were considered. Each benchmark was implemented in both Ada and FORTRAN as well as a combination of Ada and FORTRAN where appropriate. In the combination implementations the Ada program calls a FORTRAN subroutine and the FORTRAN program calls an Ada procedure or subroutine. All benchmarks were run on a VAX 11/780 and many of the benchmarks were run on a Sequent Balance 8000, and a Zenith Z-248 with an 80287 math coprocessor. Sufficient time was not available to run all benchmarks on the Sequent Balance 8000 and the pragma INTERFACE for FORTRAN was not implemented in the Alsys compiler used on the Zenith Z-248. The results of these comparisons are shown in Tables 1 and 2. Table 1 shows the execution times and Table 2 shows the storage requirements for the various implementations. As can be seen from Table 1 the execution times are similar for all implementations on the same machine but vary greatly between machines. Notice that the MATH_TASKS benchmark took longer on all machines. This is due to task switching. Each task is called several hundred times in the MATH_TASKS benchmark. The tasks in the PRIME_TASKS benchmark are only called once. Therefore, it ran in approximately the same amount of time as the other implementations. The time penalty associated with task context switching for the MATH_TASKS benchmark should disappear if run on a parallel processor architecture with each task executing on its own processor. However, allocation of tasks to processors does not happen automatically. The MATH_TASKS benchmark also took longer to run on the Sequent Balance machine even though it has a parallel processor architecture. The benchmark executed as if it were running on a sequential processor.

Table 2 shows that the object code generated by the Ada compilers is considerably more than the object code generated by the FORTRAN compilers. Programs with Ada tasks require even more object code as one might expect. As can be seen from Table 2 different vendor's compilers generate significantly different amounts of object code. This finding is consistent with an Ada research report written by the University of Central Florida (7).

Table 1 Comparison of Execution Times

Machine/Model	Implementation FOR	F/A	A/F	Ada
DEC VAX 11/780				
PRIME_PROC	30.05	33.19	30.29	33.15
PRIME_TASKS	N/A	N/A	N/A	32.59
MATH_PROC	9.07	9.87	8.87	9.97
MATH_TASKS	N/A	N/A	N/A	22.93
SEQUENT BALANCE 8000				
PRIME_PROC	94.1	N/A	N/A	184.8
PRIME_TASKS	N/A	N/A	N/A	184.3
MATH_PROC	15.8	N/A	N/A	31.0
MATH_TASKS	N/A	N/A	N/A	41.9
ZENITH Z-248				
PRIME_PROC	546.46	N/A	N/A	802.96
PRIME_TASKS	N/A	N/A	N/A	696.29
MATH_PROC	63.27	N/A	N/A	92.10
MATH_TASKS	N/A	N/A	N/A	99.74

Times are in seconds
FOR: All FORTRAN Implementation
F/A: FORTRAN Implementation Calling an Ada Procedure
A/F: Ada Implementation Calling a FORTRAN Subroutine
Ada: All Ada Implementation
N/A: Not available

PRIME_PROC: Prime Number Program with Procedures
PRIME_TASKS: Prime Number Program with Tasks
MATH_PROC: Math Program with Procedures
MATH_TASKS: Math Program with Tasks

Table 2 Comparison of Storage Requirements (Bytes)

Machine/Model	Implementation FOR	F/A	A/F	Ada
DEC VAX 11/780				
PRIME_PROC	2,764	5,592	7,570	7,464
PRIME_TASKS	N/A	N/A	N/A	8,308
MATH_PROC	2,864	3,652	4,772	4,928
MATH_TASKS	N/A	N/A	N/A	6,196
SEQUENT BALANCE 8000				
PRIME_PROC	24,576	N/A	N/A	97,292
PRIME_TASKS	N/A	N/A	N/A	120,220
MATH_PROC	14,336	N/A	N/A	18,524
MATH_TASKS	N/A	N/A	N/A	41,432
ZENITH Z-248				
PRIME_PROC	35,528	N/A	N/A	43,297
PRIME_TASKS	N/A	N/A	N/A	61,137
MATH_PROC	32,152	N/A	N/A	14,577
MATH_TASKS	N/A	N/A	N/A	33,821

FOR: All FORTRAN Implementation
F/A: FORTRAN Implementation Calling an Ada Procedure
A/F: Ada Implementation Calling a FORTRAN Subroutine
Ada: All Ada Implementation
N/A: Not available

PRIME_PROC: Prime Number Program with Procedures
PRIME_TASKS: Prime Number Program with Tasks
MATH_PROC: Math Program with Procedures
MATH_TASKS: Math Program with Tasks

Table 3 shows the results obtained from running the modified Dhrystone benchmark on all three machines. The modified Dhrystone consists of the original Dhrystone with some of the integer operations changed to floating point operations. In addition a FORTRAN version of the modified Dhrystone was prepared in order to facilitate comparison of FORTRAN and Ada code. The modified Dhrystone was originally available only in an Ada version included in a research report written by the University of Central Florida. The results obtained on the VAX 11/780 indicated that VAX Ada and VAX FORTRAN were almost identical in execution speed, with VAX Ada having a slight edge. The results obtained with the Zenith Z-248 would suggest that the Alsys Ada compiled code executes twice as fast as Microsoft's Fortran compiled code. This conclusion is at variance with the results obtained from executing the other benchmarks, which showed that the Ada programs took significantly longer to execute than the FORTRAN programs. A closer look at the figures shows even more discrepancies. For example the Ada modified Dhrystone benchmark ran at approximately three quarters the speed of the same program on the VAX 11/780, but the prime numbers program took 24 times as long to execute on the Zenith as on the VAX. Similarly, the FORTRAN version of the modified Dhrystone program took approximately one third the time on the Zenith as it did on the VAX, while the FORTRAN version of the prime numbers program took approximately 18 times as long on the Zenith as it did on the VAX. It should be noted that both the prime numbers and the math benchmarks are floating point intensive, while the modified Dhrystone does very few floating point operations. However, the prime numbers and the math benchmarks give results which can be checked for accuracy. In order to give accurate results, they must perform the same operations on all machines. The modified Dhrystone does not provide results other than timing data, and in order to verify that each step is being executed, some sort of debugging tool would be needed. No debugging tools were available for the Zenith to investigate the possibility that some modified Dhrystone steps were not being performed. Table 4 contains a list of execution times compared to the VAX 11/780 for all the benchmarks. The VAX has been assigned an execution time of 1 as a reference.

Table 3 Modified Dhrystone Execution Times.

Machine	Ada*	FORTRAN**
VAX 11/780	94,131	93,743
Sequent Balance 8000	22,740	26,061
Zenith Z-248	72,209	35,685

* Lines of Ada code executed per second.
** Lines of Fortran equivalent Ada Code executed per second.

Digital Equipment Corporation's Ada compiler version 1.1 and FORTRAN compiler version 4.6 were used. VERDIX Corporation's VADS® Ada compiler version 5.41 and DYNIX® FORTRAN compiler version 2.6 were used on the Sequent Balance running the DYNIX® operating system. DYNIX is a version of UNIX®. The Alsys Ada compiler version 1.2, and the Microsoft Fortran compiler version 3.2 were used on the Zenith Z-248.

MACHINE	PNUM			MATH			MODDHRY	
	ADA		FORTRAN	ADA		FORTRAN	ADA	FORTRAN
	TASK	PROCS		TASK	PROCS			
VAX 11/780	1.00	1.00	1.00	1.00	1.00	1.00	1.00	100
SEQ. BAL 8000	5.66	5.57	3.13	1.83	3.11	1.74	4.14	3.59
ZENITH Z-248	21.37	24.22	18.18	4.35	9.24	6.98	1.30	2.63

PNUM: PRIME NUMBER PROGRAM
MATH: MATH PROGRAM
MODDHRY: MODIFIED_DHRYSTONE PROGRAM
SEQUNT BAL 8000: SEQUENT BALANCE 8000
ADA: ADA IMPLEMENTATION
FORTRAN: FORTRAN IMPLEMENTATION
TASK: TASK IMPLEMENTATION IN ADA
PROC: PROCEDURE IMPLEMENTATION IN ADA

Table 4 Execution Times Compared to the VAX 11/780

MACHINE ENVIRONMENTAL CONSIDERATIONS

The most important consideration is that the Ada compiler being used must implement the pragma INTERFACE. Equally important the FORTRAN compiler must allow subroutine calls to and from other languages. The linker or program that generates an executable module must be able to resolve address entries and the passing of parameters. Another nontrivial consideration is differences in arithmetic speed between Ada and FORTRAN for essentially the same precision. A comparison of floating point types is shown in Table 5 for DEC[R] Ada and FORTRAN. Table 6 shows the CPU time required to compute the number of prime numbers between 1 and 100,000 using the various floating point representations. The PRIME_PROC benchmark was used to compute the prime numbers on a VAX 11/780 with a floating point processor. LONG_FLOAT took more than 143 times as much time as REAL*8 even though they are both 64 bits and essentially the same machine representation. Apparently the FORTRAN compiler uses the hardware floating point processor and the Ada compiler does not. Hopefully this difference will be corrected in a later version of the Ada compiler. As can be seen from Table 6, very high precision arithmetic takes a long time in both languages.

Ada features strong data typing of objects. However, the Ada compiler cannot check the type of a variable in another language. Hence it is easy to get erroneous results due to a type mismatch. For example, a FORTRAN subroutine can return an integer result to an object of type Float in Ada.

OTHER CONSIDERATIONS

Usually the government buys trainers with 50 percent spare execution time and main memory. Sometimes the spare capacity is less than 50 percent. Figure 1 shows the relative cost per instruction versus spare time and memory capacity. It can be seen that cost goes up considerably for additional instructions that must be written once the 50 percent spare capacity is exceeded. Cost increases for several reasons. Probably the most significant reason is that code must be written more efficiently. It may even be necessary to rewrite some of the code that was not intended to be modified in order for an upgrade to fit. Therefore, when considering doing an upgrade in Ada, a thorough timing and sizing analysis is a must. A trade off analysis should be done to determine if it would be more cost effective to change hardware to stay below the 50 percent spare capacity point.

Table 5 Comparison of Arithmetic Types

Type	Language: Ada	FORTRAN
FLOAT and REAL*4	32 bits Precision 6 decimal digits Range 0.29E-38 to 1.7E38	32 bits Precision 7 decimal digits Range 0.29E-38 to 1.7E38
LONG_FLOAT and REAL*8	64 bits Precision 15 decimal digits Range 0.6E-308 to 0.9E308	64 bits Precision 15 decimal digits Range 0.56G-308 to 0.9G308
LONG_LONG_FLOAT and REAL*16	128 bits Precision 33 decimal digits Range 0.84E-4932 to 0.59E4932	128 bits Precision 33 decimal digits Range 0.84Q-4932 to 0.59Q4932

Note: Ranges include both positive and negative numbers

Table 6 Execution Time Versus Arithmetic Types

FORTRAN		ADA	
REAL*4	00:00:26.80	FLOAT	00:00:26.60
REAL*8	00:00:39.81	LONG_FLOAT	01:35:54.54
REAL*16	01:46:43.49	LONG_LONG_FLOAT	01:54:27.35

All times are in Hours:Minutes:Seconds
Model: Prime Numbers with Procedures

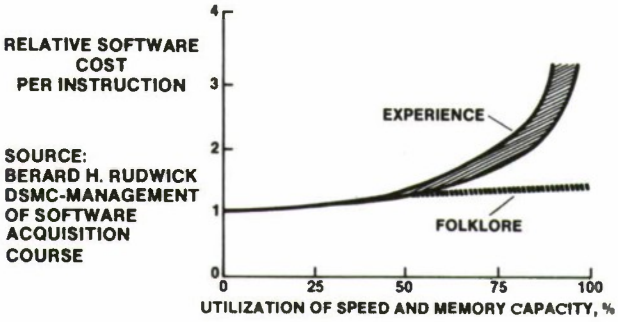


FIGURE 1. HARDWARE CONSTRAINTS VERSUS SOFTWARE COST

Benchmarks representative of the Ada code to be implemented should be run to obtain timing and sizing estimates. Differences in execution times are expected for different computers. However, this paper and others show that different Ada compilers generate significantly different amounts of object code for the same source code.

Another important consideration is that of the real time debugger to be used. Does it support the use of source code in two different languages? Of course, it is important that the debugger support the host/target environment to be used. However, this issue is independent of the high order language being used.

Life cycle cost should be given prime consideration when performing a major software upgrade. As more and more software is written in Ada, we can expect the cost of maintaining FORTRAN code to increase. Therefore, when considering a major software upgrade, the program manager should consider rewriting in Ada all of the code that is likely to change during the life cycle of the trainer.

FUTURE PLANS

In order to gain more experience integrating FORTRAN and Ada, the benchmarks will be run on several other machines. Two parallel processor machines as well as other microprocessors will be used.

Plans are currently underway to rewrite some of the Passive Acoustic Analysis Trainer's FORTRAN modules in Ada. This will allow the Naval Training Systems Center to gain practical experience integrating Ada into a FORTRAN based trainer.

SUMMARY

There appears to be no technical reason why Ada cannot be used for major upgrades of existing FORTRAN designs. Also it appears very feasible to reuse FORTRAN models and code in a new Ada design. However, some vendor's software products are easier to interface than others. All software products should improve in the future as it becomes clearer that interfacing Ada to FORTRAN and other languages is very desirable. Existing training device code should be reused when it is cost effective to do so.

The purpose of this paper is to point out issues that should be considered when planning to integrate FORTRAN and Ada. General assumptions should not be made based on the data presented. For example, it should not be assumed that Ada compilers produce twice as much object code as FORTRAN compilers. The issues described in this paper requiring consideration should be examined in the context of the planned implementation. Of all issues that should be considered, timing and sizing appear to be the most critical.

REFERENCES

1. Ada Language Reference Manual, ANSI/MIL-STD-1815A, 1983.
2. OPNAV INSTRUCTION 5200.28, 1986.
3. DOD Directive 3405.2, 1987.
4. Management of Software Acquisition Course, Defense Systems Management College, Ft. Belvoir, VA, 1987.
5. Cohen, Normal H., Ada as a Second Language. McGraw-Hill 1986.
6. VAX TM Ada Language Reference Manual, Digital Equipment Corporation, 1985.
7. Hughes, Charles E., Knowles, Henry and Lacy, Lee, "Ada Risk Assessment Report", University of Central Florida, 1986.
8. VERDIX Ada Development System Sequent/DYNIX Users Manual, VERDIX Cooperation, 1987.
9. Weicker, R., "Dhrystone: A Synthetic Systems Programming Benchmark", CACM 27,20, pp. 1013-1030, 1984.

About The Authors

William F. Parrish, Jr. is a Supervisory Electronics Engineer in the Surface/Submarine Warfare Software Branch at the Naval Training Systems Center and is currently involved in procuring trainers with Ada software. He has over 20 years of software development experience. Prior to his employment with the Navy, he was a Senior Staff Engineer with Sperry Rand Corporation. Mr. Parrish holds a BSE degree in Electrical Engineering and a MSE degree in Computer Engineering from the University of Alabama in Huntsville.

Kent G. Trewick is an Electronics Engineer in the Surface/Submarine Warfare Software Branch at the Naval Training Systems Center where he is the Software Engineer for several trainers that are being procured. One of his trainers is the HARPOON HSCLCS-1A Operator Team Trainer whose software is being written by McDonnell Douglas Astronautics in Ada. Mr. Trewick holds a BSE degree in Electrical Engineering from North Carolina A&T State University.

ADA* AND THE ISSUE OF PORTABILITY

Marianne Forsyth
Cubic Corporation
Defense Systems Division
9333 Balboa Avenue
San Diego, California 92123

ABSTRACT

Maintainable and reusable software is a benefit gained from developing training device software in the Ada program language. Furthermore, reusable simulator software can reduce development and life cycle costs. Previous languages for simulator software development such as Pascal or FORTRAN have lacked the strict standardization of a programming language like Ada. This standardization will lead to software which is more effective, more reliable, easier to maintain and reuse. A cost-saving benefit of Ada contributing to reusability is the feature of software portability. In order to use a particular Ada compiler on a simulator project the compiler must be validated by passing the Ada Compiler Validation Capability (ACVC) test suite. Validation of an Ada compiler is the process of testing the conformity of the compiler to the Ada programming language standard, ANSI/MIL-STD-1815A. However, using a validated compiler does not ensure software portability between different compilers and multiple computer systems or for that matter between different compilers on the same computer system. Implementation-dependent constructs listed in Chapter 13 in the Ada Language Reference Manual, which are tested as part of the validation test suite, provide the primary reason for portability difficulties and code incompatibility. Some compiler vendors may fully implement these Chapter 13 features while other vendors may not. In addition, the method of implementation may differ between compilers. These implementation-dependent features may be an obstacle in the benefit of portability. Although portability is considered to be an implementation level issue there are issues which must be considered during design. This paper will discuss an approach to portability, based on experience gained from the lessons learned, problems encountered and analysis performed. In conclusion, guidelines enhancing the prospect of developing portable Ada code are discussed.

INTRODUCTION

The development and maturation of the Ada programming language can promote reduction of life cycle cost and development time through the use of both reusable and portable high order language (HOL) code. The Ada language provides many features that will aid in the development of reusable and portable code that have been lacking in other programming languages such as FORTRAN and Pascal. However, some features will reduce the likelihood of producing portable and reusable code if not closely monitored and controlled during code design and development.

At this point it is advisable to define software portability and reusability to avoid any misunderstanding of the material presented in this paper.

Reusability is the capability to reuse previously developed software components in other applications using the same computer environment. The degree of reusability is measured by the extent to which a component of code, e.g., package, procedure, unit, etc., can be used in multiple applications in a computer environment identical to that of the original code implementation.

Portability, on the other hand, is the capability to transfer, with ease, previously developed software from one computer environment to another. The criteria for portability are: The software must be recompiled on a different compiler; it must perform identically under all implementations and it must produce identical performance results. The quantity of source code modification required to achieve identical functional code performance is considered the degree of portability. A perfectly portable program requires no source code modification when moved to a new target computer environment and achieves identical results as in the host environment. At the other extreme is the ported source software that must be completely rewritten due to major performance problems or different and inaccurate results. In general, software is considered highly portable if the effort required to move the source onto a new system is less than the effort to initially implement the program on the host or development system. According to John Nissen (1), the measurement of degree of portability is the fraction:

$$1 - \frac{\text{cost of re-implementation on new target}}{\text{cost of original implementation}}$$

Thus, lower cost of re-implementation results in a higher degree of portability.

A high degree of portability can substantially reduce life cycle costs after initial investment. For example, if basic simulation functions (such as instructor/operator display stations and dynamic motion and position update modeling) were developed with a high degree of portability, the development cost would be reduced on future simulation applications, i.e., no costly redesign and redocumentation. The software development schedule would also be reduced by re-using highly portable software, since only minor recoding and testing are required. Therefore, developing reusable portable software is advantageous since it reduces cost and schedule.

An organization that achieves expertise in developing portable software may elect to have a single development system and thereby further reduce development costs. There are many benefits to developing software on one dedicated system.

Software personnel can become proficient with the particular editor, operating system and tools

*Ada is a registered trademark of the U.S. Government (Ada Joint Program Office).

used for developing and testing code. Productivity will be increased by eliminating the learning curve required to become familiar with a new/different computer system. In addition a library or repository for reusable portable software components could be established on this one system.

This paper will delve into some of the understanding, concerns and potential solutions on the concept of developing portable Ada code.

HOW ADA CONTRIBUTES TO PORTABILITY

Attempts to develop portable software prior to the introduction of Ada were hindered by the different dialects of standardized languages, such as the cases of FORTRAN and Pascal. Problems emerged from attempts to move one American Standards Association (ASA) standard FORTRAN program from one ASA standard compiler to another ASA standard FORTRAN compiler. Extended versions of FORTRAN proved to be non-equivalent to the language definition which caused difficulties for portability. In pursuit of developing portable software, various methods have been attempted. Tools such as systematic translators, general-purpose macro processors, portability filters and verifiers were developed, but all have had limited capabilities in achieving true portability. Due to the lack of uniformity among higher order language compilers, efforts to translate software across different machines proved to be a time consuming and expensive effort.

Portability became a goal in the development of the Ada language, and this goal can be achieved due to three factors. The first is the standardization of the language. In December 1980 MIL-STD-1815 was established as the DOD Standard for Ada. In February 1983 the ANSI standard of Ada was granted. A 1983 revision resulted in ANSI/MIL-STD-1815A, which became the standard definition of the Ada programming language, referred to as the Language Reference Manual (LRM).

The second factor ensuring portability is the Ada trademark. Ada is a registered trademark of the U.S. Government and its use is administered by the Ada Joint Program Office (AJPO). The use of the term "Ada" indicates conformance to ANSI/MIL-STD-1815A. Ada dialects are prevented from emerging due to the fixed definition of the language which is accompanied by the trademark.

The third factor is the validation requirement before a compiler can be called an Ada compiler. Validation is the process of testing the conformity of a compiler to the Ada programming language standard, ANSI/MIL-STD-1815A. The Ada Compiler Validation Capability (ACVC) test suite is a set of Ada programs that test for this conformity. Ada compilers are validated and subsequently revalidated on a yearly basis. A compiler either fails or passes the testing; passage certifies conformity to the Ada standard but does not imply any form of product warranty or performance characteristics. The results of the testing are documented in a Validation Summary Report (VSR). A validation certificate is issued by the Ada Joint Program Office and certifies a successful test against the ACVC test suite.

Conformity testing includes checking for those characteristics that must be present in a compiler as well as for optional characteristics which must conform to the standard if they are implemented. The tested compiler may implement allowable options specified in the language standard; these options are detailed in the Validation Summary Report. The testing also identifies behavior that is implementation-dependent but permitted by ANSI/MIL-STD-1815A. These three factors result in a standardized language enforced by a trademark which is implemented on validated compilers.

Several aspects of the Ada language itself contribute to portability, such as data typing, data abstraction, generics and predefined library packages.

The Ada language provides for several predefined data types such as Integer, String, Character, Boolean, and Float. In addition, Ada allows programmer-defined types that must be explicitly declared prior to their use. User control of numerical data via data typing assists portability. The concern for indicating the number of bits or bytes or word size for different environments is eliminated. The programmer can specify the ranges on integers which is advantageous when porting code.

Data abstraction is the encapsulation of a data structure and the operations associated with that data structure. The details of the representation of data are separated from the abstract operations defined on the data. Data abstraction assists portability in that the details of the representation of data can be kept separate from the logical operations on the data. The discipline of Ada for range declarations and strong typing results in portable programs.

The generic language feature of Ada provides for the generalization of program units within the framework of strong typing. Generics allow for the solving of a set of similar but not identical problems with a single program unit. The generic program is a template for packages and programs and can be thought of as a parameterized model of program units. When formal parameters are matched with actual parameters, this is known as creating an instance or generic instantiation. Generics support reusability and portability by allowing the ease in generation of many slightly different instances.

Every implementation of the Ada language must provide the following six predefined library packages: ASCII, CALENDAR, IO_EXCEPTIONS, LOW_LEVEL_IO, SYSTEM, and TEXT_IO. These packages provide date and time utilities; facilities for dealing with errors that arise from input and output operations; for performing input and output functions; for direct control of input and output devices; provision for type declarations and constant declarations which are characteristic of the particular computer; and provides constant declarations for values of the character type which may not be available on all data-entry equipment. In the past, these facilities and utilities were provided as part of the operating system. When software that employed an operating system feature was later moved to a different machine, the software required modification and/or redesign in order to achieve the same function, execution efficiency and expected results. Fortunately with Ada these utilities are part of the language definition.

HOW ADA DOES NOT CONTRIBUTE TO PORTABILITY

The Ada Standard allows for permissible variation and specifies the manner of the variation. An example of a permissible variation is the represented values of fixed or floating point numeric quantities and the results of operations upon them. Implementation-dependencies are allowed for implementation-dependent pragmas (pragmas permit commands or special requests to the compiler), for certain machine-dependent conventions stated in Chapter 13 of ANSI/MIL-STD-1815A, and for certain allowed restrictions on representation clauses (which control how types are mapped onto the memory of the underlying machine). Acknowledging the differences in computer architecture, the Ada language allows for a set of machine constants which are implementation defined. Some of the implementation considerations are:

- Value of hardware addresses
  This type (usually integer) whose values identify memory locations. In a machine with a segmented architecture this might be a record type containing a segment name plus an offset into that segment, whereas on a distributed system this might be the name of a processor in the system plus an address into that processor's storage. (3)

- Number of bits in the storage unit corresponding to a single address
  This can be different due to machine architecture. For example, in the IBM 370 architecture this is an eight-bit byte. By contrast, in the Univac 1100 it is a thirty six-bit word. (3)

- Number of storage units

- Smallest integer that can be specified in an integer-type declaration

- Largest integer that can be specified in an integer-type declaration

- Largest number of digits of precision that can be named in a floating-point type declaration

- Smallest delta that can be specified in a fixed point type declaration

- Number of seconds in each "tick" of the machine.

These machine characteristics are provided in the predefined package known as SYSTEM. These low-level programming features refer to the underlying machine hardware, so their meaning varies from one compiler to another. These features allow the program to execute specific machine instructions including device-level input and output operations. Ada allows an implementation considerable leeway in defining the allowable forms of low-level features and their meaning for a particular machine.

One purpose of the compiler validation testing is to determine that the implementation-dependent behavior is allowed by the Ada Standard. The implementation-specific values required for such tests are listed in Appendix C of the Validation Summary Report. In addition, every Ada implementation supplies an Ada Language Reference Manual, which is the implementation's representation of ANSI/MIL-STD-1815A. The implementation-dependent characteristics of the compiler are included in Appendix F of this manual. Therefore, it can be said that machine dependencies are allowed in a controlled manner by the Ada Standard through the validation process and documented by each implementation.

The concept of portability is weakened by these implementation-dependent features. There is a risk associated with portability due to different computer architectures. For instance, Ada software developed on a compiler for a machine with 36-bit words may prove difficult to port to a system with 32-bit words. The allowance of implementation-dependent pragmas also can hinder portability. A pragma that is not implemented on a target system is ignored by the compiler and no indication or warning is required by MIL-STD-1815A. Also the danger of porting code which uses non-standard pragmas is exemplified when the target system may have a pragma with the same name as that of the host system but has a different implementation. The target compiler will not be able to detect this, although both systems could have passed validation of conformity for implementation-dependent behavior. Thus, portability is not ensured by validation. Problems and difficulties can still arise when porting Ada software from a validated host system to a validated target system.

PORTABILITY EXAMPLES

Experience with portability is based on developing Ada code on a Digital Equipment Corporation (DEC) MicroVAX II Computer utilizing the DEC Ada Compiler and then porting the code to a Gould computer with a Telesoft compiler and Masscomp 5600 computer with a Verdix compiler. Table 1 provides a description of the computer systems.

A fairly simple non-real-time software program was developed on the MicroVAX II and ported to the two target systems with no modifications required, indicating that portability can be achieved with the Ada language.

A non-real-time software costing model also was implemented and is more characteristic of problems to be encountered. A line of code estimate would be prompted for and then the software program would calculate the man-month effort and time required for development. Difficulty in porting this application arose from an implementation consideration. The Ada language provides mathematical operations for addition, multiplication and division as part of the language standard. The implementation must provide the library routines to perform square root, logarithmic, exponential, sine, cosine, tangent, arc sine, arc cosine and arc tangent functions. Access to these library routines are accomplished by the WITH clause in the program unit which requires such visibility. A logarithmic calculation was required so that the interface package to the VAX/VMS RTL Mathematical Library routine, called MATH_LIB, was imported by the WITH clause. Also the instantiation of a commonly used generic package, called FLOAT_MATH_LIB, was imported by the WITH clause and direct visibility indicated by the USE clause. When moved to either target system, the software did not compile, naming as the error a nonexistent

TABLE 1

	Gould	Masscomp	DEC
Model #	pn 6031	5600	MicroVAX II
Ada Compiler Version #	Gould (Telesoft) Ver. 3.08b0	Beta Release Ver. 5.41	VAX Ada Ver. 1.0
Operating System	UTX/32, Ver. 2.0	RTU 3.1	VAX/VMS Ver. 4.4
Version #	Rel. U01		
Internal Memory	8 Mb	4Mb	9Mb
Cache	8 kb	8kb	None
Load Module Size Limit	64 kb	N/A*	1 Gb
Rated MIPS	1.7	2.5	1.0
Optimizer Options	None	None	Time, Space, Run-time Check
Single/Multi CPU	Single	Single	Single

*Masscomp uses a swap file. Limit of swap file on system used was 16 Mb.

library unit. Remember that the criteria for portability is to be recompiled on a different compiler, perform identically under all implementations, and produce identical performance results. In order to get successful compilations on either target, the source code had to be modified to that particular implementation's correct math package names. As in the case of the Telesoft compiler on the Gould, the names were modified to GENERAL_MATH, and LONG_FLOAT_MATH. Once this application on the target systems compiled, exact results as on the MicroVAX II were obtained. As previously mentioned, implementation considerations weaken the concept of portability. Therefore when using an implementation's packages (other than language defined like TEXT_IO, CALENDAR, etc.) one must be aware that source code may, and probably will, have to be modified to accommodate target implementation naming conventions.

A software development which is still in progress involves a real-time application of a target math model involving a user interface. The predefined Ada package TEXT_IO is being used although it is "wait input/output". To emulate "no wait input/output" of an input from a terminal keyboard a VAX system service called, QIO, is being utilized. This software component has been isolated in a separate package, and all software team members are aware that when this application is to be moved to the target systems, modifications will be necessary. This situation was recognized early in the development cycle and the design accommodates this implementation consideration.

PORTABILITY GUIDELINES

Portability is enhanced when more facilities are provided in the language as in the case of Ada. The degree of portability to be achieved is influenced by the software design. The approach should be to avoid machine-dependent considerations when designing the high levels of programs. The machine dependencies should be taken into account at the detail design phase. As discussed previously, Ada allows for certain implementation-dependent features which could hinder portability. Avoidance of such features may ensure a higher degree of portability but may not be feasible due to real-time application requirements. Portability can be enhanced by the logical and physical isolation of implementation dependencies through use of packages. Therefore, when porting the software, the implementation-dependent part is kept logically separate, aiding in locating where modifications must take place. As described earlier, this is the approach being used in the real-time application requiring a system call.

When the target system is identified prior to development, Appendix F of each implementation's representation of ANSI/MIL-STD-1815A is important reference material to ensure portability between the systems. The specification of the predefined package SYSTEM is in this appendix. It contains the machine characteristic values discussed earlier, such as value of hardware addresses, number of seconds in each "tick" of the machine, etc.

If a component must refer to a machine dependency, it is best done in terms of the package SYSTEM. For example, declaring a floating point type to a system's maximum precision such as:

```
TYPE Scale_Type IS DIGITS 9;
```

would be appropriate in the case of a Masscomp 5600 with a Verdix compiler. However, if this code was ported to a Gould system with a Telesoft compiler, a problem would occur because this system does not support nine digits of precision, (the maximum precision is six). Instead if expressed in terms of the package SYSTEM portability can be ensured by making the declaration

```
TYPE Scale_Type IS DIGITS System.Max_Digits;
```

Other implementation-dependent characteristics, such as implementation-dependent pragmas, are defined in this appendix. What is/is not implemented on each system can influence the degree of ease there will be in porting the software.

When the target system is known, it also is advisable to organize a porting team from selected software team members. The porting team's responsibility is to make certain that implementation considerations are taken into account during the design phase. The porting team must be aware of any differences in the two computers' architectures and become familiar with Appendix F for each implementation. The actual movement and implementation on the target system is done by this team.

The third recommendation is the reading of the book, Portability and Style in Ada, edited by John Nissen and Peter Wallis. This useful guide is organized in the format of ANSI/MIL-STD-1815A and proposes a set of rules or recommendations to aid portability for many sections of the standard.

CONCLUSION

In conclusion, this paper has shown that the Ada language has many strong features that assist in the development of reusable and portable code. Ada is a controlled language by the adherence to ANSI/MIL-STD-1815A; furthermore, the trademark assures conformity to this standard and required compiler validation. At the same time, differences in machine architecture has dictated the allowance of machine characteristics and implementation dependencies. This paper has shown that these features weaken the concept of portability and must be carefully considered in the design phase of software development. Awareness of the differences that may exist between development and target systems and accounting for these differences in terms of portability can influence the degree of success in porting Ada software.

REFERENCES

1. Nissen, John and Wallis, Peter; Portability and Style in Ada, Cambridge University Press, 1984

2. Ada Language Reference Manual, ANSI/MIL-STD-1815A, 1983

3. Cohen, Norman; Ada as a Second Language, McGraw-Hill, 1986

4. Booch, Grady; Software Engineering with Ada, Benjamin/Cummings, 1983

5. Brown, P.J.; Software Portability, Cambridge University Press, 1977

ABOUT THE AUTHOR

Marianne Forsyth is a Senior Programmer Analyst in the Defense Systems Division of Cubic Corporation. She is currently responsible for Internal Research and Development activities on Ada software development. Previous experience includes software lead positions in charge of developing real-time simulation software for operator and maintenance trainers. She holds a Bachelor of Science degree in Business Administration from San Diego State University.

REQUIREMENTS DEFINITION FOR ADA-BASED TRAINING SYSTEMS

by

Mr. Mike Caffey
and
Dr. Matt Narotam

Burtek, Inc.
Tulsa, Oklahoma

ABSTRACT

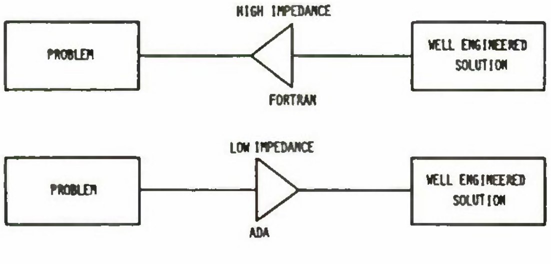
The importance of the requirements definition stage in developing Ada for simulator systems is one of the "lessons learned" on the Ada Simulator Validation Program (ASVP). The traditional approach to requirements definition generally utilized for training systems is reviewed and some of the problems that result are discussed. The types of requirements that impact the design and life cycle support of the system are defined because of their significance to the process utilized for developing the system design. The impact that Ada and Object-Oriented Design implementations have on the requirements definition process is examined by first addressing the characteristics and features of Ada that satisfy software engineering concepts. Next, the decomposition and design procedures of the method and the manner in which requirements are utilized for generating the software design are discussed. The process involved in the establishment of these requirements is also discussed. Finally, activities related to the systems requirements review process are addressed.

1.0 INTRODUCTION

Ada was developed to support the engineering approach to software development. Ada features such as tasking, packaging, generics, English-like syntax, strong typing, exception handling, etc., support software engineering concepts such as abstraction, information hiding, modularization, and generalization. The appropriate use of these features will lead to well-engineered software systems that are reliable, easily maintainable, highly reusable, and efficient. Languages such as FORTRAN provide very few features that support software engineering. This has the effect of "distancing" the problem from the desired solution. Traditional software engineering practices that are used for developing FORTRAN software therefore do not support a mapping of the program solution to the problem domain. The diagram below attempts to demonstrate that FORTRAN provides a high impedance path, whereas Ada provides a low impedance path to developing well-engineered solutions.

Ada, therefore, has an impact on traditional software engineering.

Having developed a language that will support the engineering of a solution that will map onto the problem domain, a process or method is required to handle the transition of the problem onto a solution. Burtek has refined the Object-Oriented Design process to manage this transition. This process, called Object-Oriented Development[1], employs the concepts of abstraction, information hiding, modularization, and generalization to convert problem requirements into a solution.



Object-Oriented Development possesses certain inherent characteristics which affect requirements definition. Both Object-Oriented Development and Ada place great emphasis on the requirements of the systems to be developed. This is consistent with a systems engineering approach to the development of training devices. The specification of requirements, therefore, becomes even more critical because of its impact on software design and performance. This paper addresses the impact of Ada on this process based on experience gained from the ASVP. The ASVP is a research and development contract issued by and sponsored by the USAF Aeronautical Systems Division (AFSC), Wright-Patterson AFB, Ohio. The ASVP involves the redevelopment of software for demonstration on a C-141B Operational Flight Trainer (OFT). The development effort is focused on gathering engineering design and acquisition data that highlights the effectiveness of software engineering concepts permitted by the Ada language. Part 2 of the paper briefly describes the traditional requirements definition process and the problems resulting from this process. It will be seen that this process, which influences the development of the software system, does not support Ada software development using software engineering concepts. Part 3 describes the impact of requirements on the establishment of the system design using the Object-Oriented Development method and Ada. Part 4 describes the requirements derivation process, beginning with user requirements. It is important to note that the user requirements comply with the characteristics established in Part 3. Part 5 addresses the review process and considerations which impact the review of requirements and their inclusion in the system design. Note that in the discussion that follows, the term component represents an object.

2.0 TRADITIONAL REQUIREMENTS DEFINITION PROCESS

Figure 1 depicts the traditional process for software development. Four major steps are identified beginning with requirements definition and ending with code, integration and test. The requirements definition step leads to the establishment of a data base of systems requirements. These requirements are derived from

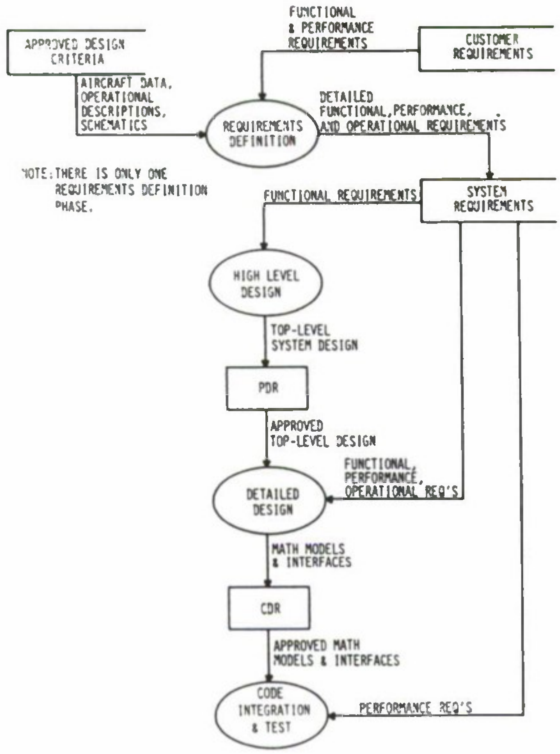


Figure 1. Traditional Requirements Definition

an approved design criteria for the system being simulated and customer requirements. The customer requirements are mainly incomplete functional and performance requirements. The systems requirements are refined and elaborated by the manufacturer.

The intermediate step results in the development of the high level design followed by detailed design of the system. There are typically two major design review phases - Preliminary Design Review (PDR) for reviewing the high level design, and Critical Design Review (CDR) for reviewing the detailed design of the system.

Traditionally, the design reviews tend to focus on implementation details instead of considering whether performance requirements have been established and that these requirements are being met. This focus on implementation detail begins at the Request for Proposal (RFP) stages where often the specification of the "system requirements" is in the form of a preliminary breakdown of system functions which tend to form the basis of the work structures and breakdown. That, together with a "pre-set" mentality of what the customer wants (a cockpit procedures trainer or an operational flight trainer or a weapons system trainer) appear to be the prerequisite for establishing the system requirements. This process tends to lead to the definition of the design using a functional approach. The functional approach elaborates the subsystem definition based on available information about functional requirements. This approach leads to ill-defined components and interfaces, resulting in software not easily maintained or reused, and that which requires several changes to account for omitted requirements during the design phase.

The traditional requirements definition process gravitates toward a specification of implementation concerns rather than focusing on a better definition of the problem scenario. This process requires modification to accommodate the procedures required to support Ada software development. The changes impact the process of requirements derivation, the types of requirements, and the characteristics of these requirements.

3.0 Ada AND DEVELOPMENT METHODS

Ada has been designed to solve many problems inherent in older programming languages. Ada constructs allow the system architecture to be more closely tied to requirements. Used appropriately, cost effective systems can be developed that are reliable, portable, and maintainable.

One major feature of the language is that software specifications can be developed and integrated to form the software structure, prior to coding. This feature supports the recursive decomposition of system into subcomponents, with well defined interfaces that can be used to generate Ada specifications. This process, supported by a Program Design Language (PDL) Tool, can provide a powerful mechanism for development of well structured and well designed software systems because it allows prototyping as the software is composed.

Object-Oriented Development best suits Ada as a development method. The packaging and tasking constructs of Ada allow the system to be partitioned into logical structures. This feature supports the process of Object-Oriented Development for generating the software structure. Object-Oriented Development allows the breakdown of the system into objects that act on other objects within the system. These objects can be implemented in Ada to develop real-time software systems. The object content of the system is determined from requirements imposed on the system and the approved system design criteria.

Figure 2 depicts the main steps in Object-Oriented Development. As shown, Object-Oriented Development is a recursive process that generates the structure of the software system. Steps 2 and 3 are recursed for "complex" subcomponents until the lowest level components are identified. Figure 3 provides a more detailed view of decomposition process. The process utilizes top level requirements to define system components and actions performed on the components. Additional requirements called "derived requirements" are generated for defining lower level components and their interactions.

The use of Object-Oriented Development for Ada software development requires design requirements to be hierarchically derived from the consideration of user requirements and the system design criteria. Top level requirements will be rather general, while lower level requirements will be somewhat more specific. Requirements pertaining to a particular component should specify only the external characteristics of the component, leaving the internal design of the component to the engineer. At the lowest level, the requirements should completely define the characteristics of the component behavior.

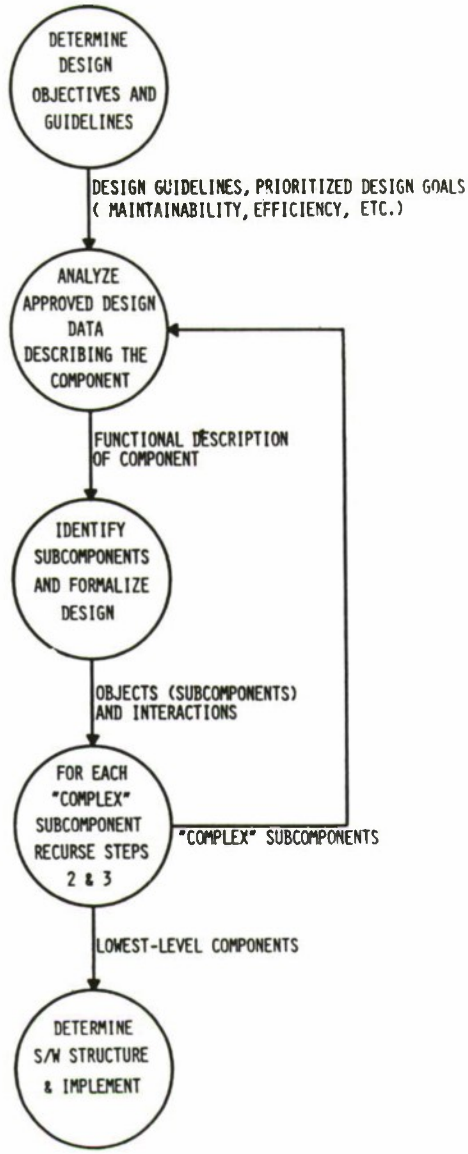


Figure 2. Main Steps in Object-Oriented Development

In addition, to address the life-cycle of the product, the requirements should specify a prioritized set of design guidelines addressing maintainability, reusability, portability, and run time efficiency, as well as outlining forceable enhancements to the product. A set of implementation priorities should be specified and the environment in which the product is to operate should be characterized.

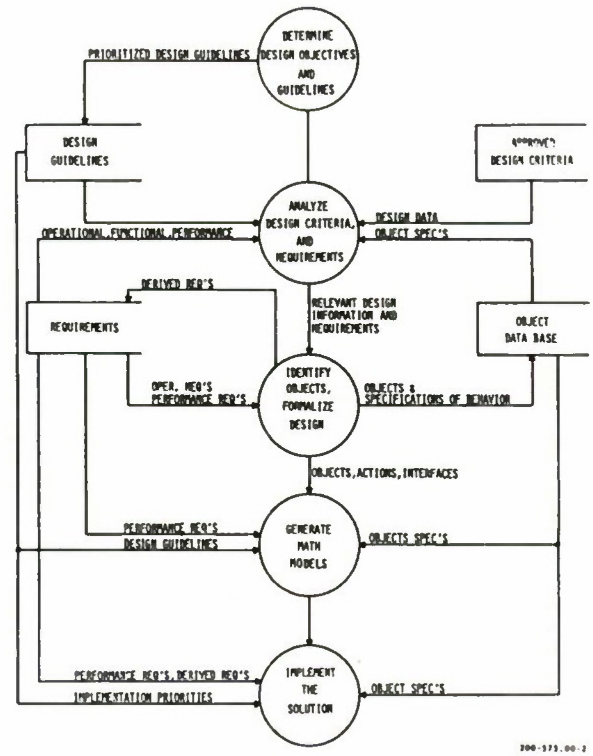


Figure 3. Detailed Decomposition Process

When specifying requirements for a real-time system, the engineer should specifically define the following:

1. Activities that should appear to be continuous or concurrent with other activities.

2. The response of the component to undesirable events.

3. Activities that are time dependent vs. activities that are purely event driven.

4. Fidelity requirements such as screen resolutions and output tolerances.

5. Required execution speed and allowable memory space.

4.0 REQUIREMENTS DEFINITION PROCESS

Figure 4 depicts the requirements definition process embedded in Object-Oriented Development. The initial requirements, combined with the approved design criteria, are used for establishing a description of the system specification in terms of components and actions using abstraction and information hiding. This initial set of requirements must be correct and sufficient to allow the definition of the system specification. These requirements are generally established by the customer by examining the training requirements of the simulator. It is essential that the customer understand the impact of these requirements on the design process. Good definition of requirements will lead to a well designed software system that is easily maintainable and usable (and reusable).

At each decomposition step, additional requirements are derived which result from consideration of limitations, assumptions, and additional design detail. The performance characteristics and fidelity of implementation will impose requirements on the system design. Constraints to the system design may be imposed by hardware or other components.

The derived requirements are utilized for generalizing low level system/subsystem specifications. These specifications form the basis for determination of components, actions, or operations to be performed by the components. This process is repeated until the system is fully defined, based on the elaboration of requirements.

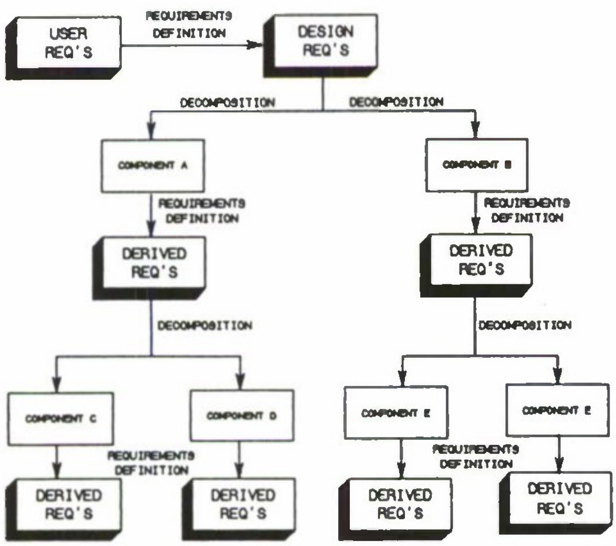


Figure 4. Requirements Definition Process

5.0 REQUIREMENTS REVIEW PROCESS

The use of Object-Oriented Development and Ada will impact the review processes during the development phases of the project. Requirement review should take place frequently until the design is complete. Working reviews should replace formal reviews. The customer must be actively involved in the review process with the objective of "working together" towards system implementation. This approach to software development results in the synthesis of the system[2], and yields all requirements, constraints, limitations, and assumptions which must form part of the life cycle support documentation. By using an appropriate design tool, the system design can be released incrementally for review and acceptance, to form the baseline for production of software. The customer can also be involved in this process, which will result in a more cost effective review and verification process.

6.0 CONCLUSION

This paper has attempted to describe the importance of requirements definition to the software development process. The process has been enhanced because of the need to apply a development method for generating Ada software. Enhancement to the process will result in a closer mapping of the software solution to the training requirements. This is dependent on well defined requirements for the training system. This puts a burden on the customer to train personnel to understand the impact of these requirements on system definition.

REFERENCES

1. Narotam, Saib, Layton; Object-Oriented Development of Training Systems in Ada, Proceedings Nov 18-20 1986, Interservice/Industry Training System Conferences, Volume 1, Page 13.

2. Hoog; Systems Engineering For Training Systems - A Team Approach, Proceedings, Nov 18-20, 1986, Interservice/Industry Training Systems Conference, Volume 1, Page 190.

ABOUT THE AUTHORS

Mr. Mike Caffey is a software systems engineer at Burtek where he develops simulation models for on board aircraft systems. Mr. Caffey holds a degree in Electrical Engineering from the University of Tulsa. While involved in the Ada Simulator Validation Program, Mr. Caffey has performed extensive research in design methodologies and tools that are used for the development of real-time software in Ada. Mr. Caffey is currently involved in the use of object-oriented design for the redevelopment of the aircraft systems software for a C-141 Operational Flight Trainer.

Dr. Matt Narotam is a staff engineer at Burtek, responsible for the technical performance of the Ada Simulator Validation Program. Dr. Narotam holds a Ph.D from the University of Salford, England, for a Thesis on Continuous Systems Simulation Language (CSSLs) implementation on mini-computers. Prior to joining Burtek, Dr. Narotam held the position of Research Scientist at the Computer Simulation Center, University of Salford, where he investigated simulation techniques for CSSLs. At Burtek, Dr. Narotam led a team of systems engineers in the development of training equipment for the F/A-18 aircraft. Prior to his appointment as staff engineer, Dr. Narotam was Supervisor, Software Systems. Dr. Narotam is an Adjunct Professor of Software Engineering at the Maths and Computer Science Department, University of Tulsa.

THE NEXT GENERATION OF TRAINERS: LESSONS LEARNED FROM THE Ada SIMULATOR VALIDATION PROGRAM

Jerry H. Hendrix

Boeing Military Airplane Company
Simulation and Training Systems
Huntsville, Alabama

ABSTRACT

The transition to the next generation of Aircrew Training Devices (ATD) is upon industry and government. More sophistication of aircraft systems, radar equipment and technical delivery systems will make simulation even more complex. Emphasis will be taken away from classical flight dynamics, atmosphere, etc. and transition toward the more complex voice recognition weapon delivery or "Darth Vader"-like helmets which allow pilots to aim weapons by turning their heads. Another transition to aleviate these problems of sophistication is Ada.*

The Ada language has been adopted by the Department of Defense for use on all mission-critical applications. Early in 1987, the Tri-Services made it clear that training systems simulations shall be in Ada. But, mandating Ada is not enough. Industry must take actions to prepare for a new transition crisis: FORTRAN "mindset" to Ada "mindset" (procedural-oriented design vs. object-intensive design).

The use of Ada and its capabilities and attributes promises to reduce the cost and increase productivity in the development life cycles. This paper discusses aspects of building real-time systems in Ada from a lessons-learned viewpoint for rehosting an existing flight trainer. Most contemporary flight simulators have been written in FORTRAN, whereas the future promises flight simulators written in Ada. As was done with FORTRAN in the past, there must be software guidelines followed when doing real-time Ada. For the most part, in the immature Ada world, the power of the compiler and computer dictates one's choice for these guidelines.

This paper discusses methodologies, compilers and guidelines of the real-time Ada software produced for the Ada Simulator Validation Program (ASVP) **. The characteristics of a real-time, Ada program can be equal to, if not better than, FORTRAN. One must realize it is extremely difficult to produce a real-time, maintainable, reuseable and loosely coupled Ada system. One can produce a reuseable and loosely coupled system, but maintainability is sacrificed. One can also build a reuseable and maintainable system, but may lose visibility control in some areas. There are many different methods and approaches for producing Ada code. Some are and some are not useable for building a real-time system.

This paper discusses the advantages and disadvantages of building a real-time Ada system using the methodology Boeing adopted on the ASVP. Comparison of FORTRAN and Ada is represented, but the emphasis is more on real-time Ada.

WHY Ada?

The Department of Defense (DoD) directive 5000.Ada mandates the use of Ada on all weapon system acquisitions. Ada was chosen by the DoD to address the existing software crisis. The language will improve software consistency, reliability and maintainability, improve productivity and reduce life cycle cost. Use of Ada allows enforcement of sound engineering principles. Those principles are: abstraction -- extraction of essential details into an understandable unit; information hiding -- details of an implementation or abstract made inaccessible; modularity and localization -- grouping logically related items and completeness where no essential details are missing.

In evaluating the use of Ada in flight simulation, one quickly realizes the need for a precise method to handle the mass amounts of data in a simulator. Simulators are data-intensive devices -- flight data, ground station data, weapon or tactical data, etc. (A misunderstanding, or nonchalant attitude, toward interfacing in Ada will lead to ambiguous and undefined systems; thus leading to a system which will not compile without large changes in the design). (There is a fundamental limit upon the complexity of which a human can cope). The basic problem in today's software design is not the mismanagement of technology, but rather the inability to manage the complexity (interfaces) of large systems. Ada provides a vehicle for defining interfaces in the design of an abstraction. The method chosen for a flight simulator design in Ada must fit the domain to:

1) Manage the complexity -- dealing with the large amounts of data.

* Ada is a registered trademark of the U.S. Government Ada Joint Program Office (AJPO).

** Funded by Contract F33657-86-C-2072 from USAF Aeronautical Systems Division (ASD/YWB)

2) Alleviate informal communication and provide enforceable communication.

3) Provide uniformity or consistency in the design.

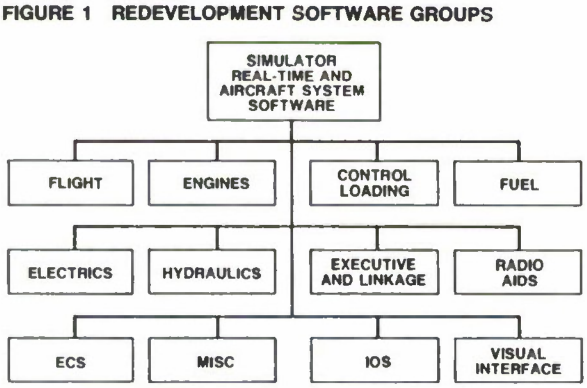
4) Allow a conceptual model of the problem space -- visualize the system environment.

On the ASVP, Boeing chose a methodology consistent with development of a large real-time system in Ada. The proponents of a traditional functional decomposition approach, was realized early in the project to lead to an unmanageable system.

ASVP SCOPE AND WORK

The ASVP is contracted research and development to investigate the applicability of using Ada in a real-time device. A simulator was chosen due to its complex man-machine interface and real-time requirements. Some flight simulators require iteration rates upwards of 60 Hertz in order to match the critical flight fidelity of the simulated aircraft. Hardware to software signal conversion equipment interfaces reach over 10,000. If Ada can operate under these strenuous requirements, it can alleviate many software problems in today's maintenance environments. The simulator chosen by Boeing had approximately 80,000 lines of code (Assembly language and FORTRAN) of which approximately 85% of the code was to be redeveloped in Ada. Figure 1 lists the types of software which were redeveloped completely in Ada. The ASVP objectives were to gain metrics on real time comparisons, gain knowledge on Ada methods and methodologies, apply the methodology to a development of a training device, record concerns and lessons learned in each phase of the development cycle and gain knowledge in the production of a large complex real-time system in Ada. The contract is a 19 month program nearing completion and Boeing has two subcontractors;Gould, supplier of the Aplex Ada Compiler, and Science Applications International Corporation (SAIC) of Huntsville for technical support and consultation.

FIGURE 1 REDEVELOPMENT SOFTWARE GROUPS



ORGANIZATIONAL STRUCTURE

At the beginning of the ASVP, the local organizational structure was a full classical matrix. During the early project phases, the structure to support ASVP became different than the matrix organization. The most important lesson learned dealing with organizational structure is the responsibility of each group. Normally, the systems engineering organization defines the system interfaces in Interface Control Documents (ICD) or Software Requirements Specifications (SRS). Early in ASVP it was noted that, in Ada, this is a redundant step. At compilation time the interfaces are made consistent. The team organization merged systems engineers and software developers into one discipline. The ASVP team was a project-oriented group with matrix support only from Project Control and Scheduling and Functional Test. The group members resided together in one work space to alleviate non-Ada distractions.

There has to be a management commitment to an Ada project. First, the manager must provide adequate resources for training, etc. He must provide adequate checks and controls during project development. Finally, he must be willing to accept little progress early in the program. The pure nature of an Ada development leads to a non-modal design. It is extremely hard to notice movement, and when noticed it is hard to measure. An item which appears to have taken an excessive amount of time to produce may later be a cornerstone in faster development in lower abstraction levels.

ASVP METHODOLOGY

The Boeing ASVP methodology chosen was a top-down, object-abstracted, tier-level development approach using Structural Analysis for requirements definition. The method enforces sound software engineering principles and the system design in Ada Program Design Language (PDL). The methodology was augmented by systems engineering knowledge in real-time software systems development. The Boeing ASVP methodology was developed for real-time use and adapted for a flight simulator discipline. The structural analysis provided a tool which helped in identification of data and the operations and states of that abstraction. The method enforces a tier-level design, code and test applied iteratively at each tier level. The design is conical in nature, enforcing the structural analysis at each level. The software phasing consists of:

- Acquisition of design criteria and data
- Analysis of design criteria
- Data flow analysis of design criteria
- State transition analysis of the data abstraction
- Layout structure in Ada PDL
- Coding of software unit
- Discrete testing of the unit
- Integration testing of the unit.

DEVELOPMENT PHASES IN ADA

The classical software development phases as described in DoD Mil-Std 2167 are: requirements, design, code, test, integration and acceptance. It was not a requirement to follow 2167 guidelines for each phase but rather to gather information about each of these phases in an Ada development.

Requirements

Normally, the System Requirements Review is 30 to 60 days after contract award. At that time, a consensus is supposed to be reached on the definition of requirements for the program. In the past, these reviews have defined specific requirements which may or may not be fully met by the contractor. Requirements Traceability Matrices (RTM) are set up to ensure the transition of requirements to design. In the past, this has never fully happened. Ada now offers a new means of dealing with requirements analysis and allocation. Figure 2 shows an example of requirements analysis in Ada. Notice that the requirements correlate directly into Ada code, and thus compilable and consistent. There are no absolutes, even with Ada. An understanding of the requirements still must be discussed and unknown or vague requirements explicitly understood. There is still a place for the RTM in an Ada design. The definition of requirements in a single or multiple Ada package makes those requirements consistent and thus affords the allocation of those requirements using the data flow analysis to identify or allocate subprograms or packages. The requirements phase is not being fully investigated on the ASVP due to the nature of the redevelopment.

FIGURE 2 REQUIREMENTS IN ADA

```
/pe STEADY_STATE_WIND_CONTROL IS
    (STANDARD_ATMOSPHERE_WINDS_ARE_SELECTED,
    STEADY_STATE_WIND_CHANGES_ARE_SELECTED,
    STEADY_STATE_WIND_LAPSE_RATE_CHANGES_ARE_SELECTED,
    FAA_WIND_PROFILES_ARE_SELECTED,
    GLOBAL_WIND_NORMAL);

 pe FAA_APPROVED_WIND_SHEAR_PROFILES IS
              (NEUTRAL_LOGARITHMIC,
              FRONTAL_1_TOKYO_1966,
              THUNDERSTORM_FAA_MATHEMATICAL,
              FRONTAL_2_LOGAN,
              THUNDERSTORM_2_PHILADELPHIA,
              THUNDERSTORM_3,
              THUNDERSTORM_4,
              THUNDERSTORM_5,
              FRONTAL_3,
              THUNDERSTORM_6_JFK);
ubtype FAA_WIND_SHEAR_PROFILES_HEAD_AND_CROSSWINDS IS
     FAA_APPROVED_WIND_SHEAR_PROFILES range NEUTRAL_LOGARITHMIC..
     FRONTAL_2_LOGAN;
ubtype FAA_WIND_SHEAR_PROFILES_HEAD_CROSSWIND_AND_VERTICAL IS
     FAA_APPROVED_WIND_SHEAR_PROFILES range
     THUNDERSTORM_2_PHILADELPHIA..THUNDERSTORM_6_JFK;
```

Design And Code

These two phases represent the most dramatic shift in the software development phase while using the Boeing methodology. The software transition is no longer from design-to-code, but from requirements-to-design/code. In the methodology, the design is laid out in Ada PDL and compiled at each tier level. The design is code and the code is compilable, thus making the design consistent. In Ada, the interfaces can be enforced if the methodology allows. Boeing defined the interfaces in Ada code and compiled the design, thus making the interfaces consistent. The interfaces are identified by using data flow analysis for interface definition. The design continues by using control flow analysis to identify the state in which a system may reside. Figures 3(a-c) give an example of a design in Ada using structural analysis. If a systems designer can draw the state transition diagram, he fully understands this problem domain. In order to augment the design the software development plan and the software Standards and Procedures manual were developed to assist designers in consistency. These documents must be the focal point for a consistent design in Ada. Each company must develop these due to the different problem nature.

FIGURE 3-a RADIO ALTIMETER DESIGN EXAMPLE

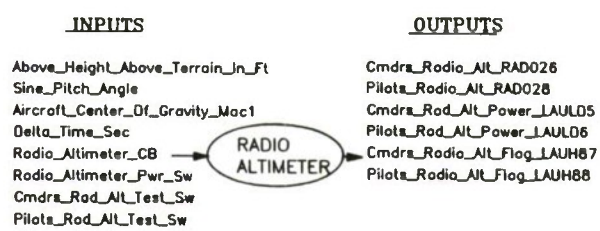


FIGURE 3-b RADIO ALTIMETER DESIGN EXAMPLE

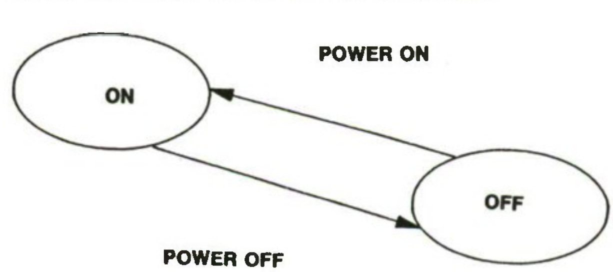


FIGURE 3-c RADIO ALTIMETER DESIGN EXAMPLE

```
begin
  Radio_Altimeter_Power := CB_State
    (Radio_Altimeter_CB)=Energized and Switch_Status
    (Radio_Altimeter_Switch)=Latched;
  if Radio_Altimeter_Power then
        radio_altimeter_power_on;
  else
        radio_altimeter_power_off;
  end if;
end radio_altimeter;
 update_radio_altimeter;
```

Software Integration And Test

There is not a classical phase called software integration (SWI) in Ada. SWI is nebulous in Ada because the SWI is handled at compilation. When something is compiled in Ada, it is integrated with that system. Once a tier has been designed/coded, testing of that unit is started. Boeing used a two-segment

testing approach: White Box Test and Black Box Test. White Box testing is used to check outputs from a unit while stimulating that unit with known inputs. A unit is the lowest level of abstraction in this application. After successful White Box testing the unit is integrated with the rest of the system and Black Box testing begins. Black Box testing is stimulating a unit while it is integrated by looking at inputs and how that unit responds to those inputs. Comparisons are then made between Black Box and White Box test results.

Maintainable, Reusable, Portable And Real Time

At implementation of a software component, one chooses whether or not to make the component reusable, maintainable or portable. One thing to note are the trade-offs between each of these Ada goals. In addition, on ASVP, the aspect of real time was examined.

Maintainability is the ability by which a component is easily changed and supported over a life cycle. Since maintenance is a relative thing, it must be noted that endless discussions can result over whether a component is maintainable or not. The largest discussion is whether a component will deteriorate with change (lose clarity). The building of maintainable software must be designed early in the project. Trade-offs exist at every level. For example, the ASVP design contains a package which includes all interfaces to lights in the cockpit. In that package are two functions which are apparent to the outside users. TURN_ON_THE_LIGHT and TURN_OFF_THE_LIGHT. In addition, the definition of all lights are made visible in this package. Visibility is the amount of accessibility into an implementation or abstraction from outside components. Immediately, attention is brought to the declaration of all lights in one place. The system designer of the fuel system, for example, may turn on an electrical light! Yes, but because the lights are together lends itself to an ease in maintenance and diagnostics. If a new light is added to the cockpit, there is a logical place to input that new light in the software package. In diagnostics, it is now easy to determine if there is a software or hardware problem. No longer are large diagnostic programs needed. Figure 4 shows a reusable diagnostic program for the lights application.

FIGURE 4 DIAGNOSTIC LIGHT ROUTINE

```
with Lights_Package;
procedure Test_the_Lights is

begin
  Lights_Diagnostic_Loop: for Light in Lights_Package.
    Light_Name loop
      Lights_Package.Turn_on_the_Light (Light);

      delay 2.5;
      Lights_Package.Turn_off_the_Light (Light);

      delay 2.5;
    end loop Lights_Diagnostic_Loop;
end Test_the_Lights;
```

Reusability can have dual meaning; reusability in the large (across project to project) or reusability in the small (uses within a project). One must note that reusability must be designed into the development just as is maintainable software. Early in the development, components deemed reusable must be planned. Normally, building of reusable components leads to an increase in computational execution speed.

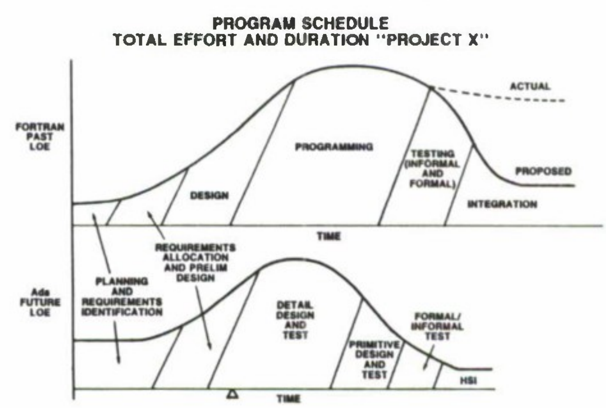
Portability is the ability to transfer programs from computer to computer and allow compilation and execution. The ASVP design is completely portable except in three areas. These three areas are all real-time, specific areas. The use of math routines are Gould-specific. The routines were written in assembly language to increase execution time. Second, the interrupts used to trigger a new frame were operating system dependent. Calendar clock resolution was only 20ms and would not afford the resolution needed to drive the simulation at its executable rate. Finally, High Speed Devices (HSD) were used to transfer data to and from linkage interfaces. The HSDs allowed a no-wait input and output.

PRELIMINARY DEVELOPMENT METRICS

Manloading

Figure 5 shows a comparison of level of effort for a typical program in Ada and in FORTRAN. The data was extracted partially from manhour reporting on ASVP. The chart displays trends and not absolutes.

FIGURE 5 LEVEL OF EFFORT COMPARISON



Ada vs FORTRAN

Figures 6 - 8 show comparisons of compilation speed, memory usage and execution time for Ada on our compiler. The machine and compiler used on the ASVP were Gould 97/80 (Dual Processor) with the Gould APLEX compiler.

The resultant of Boeing's methodology and the use of Gould's compiler lends itself easily to producing efficient real-time code. Early in the project, it was decided to design the system independent of the overhead for subprogram calls. As execution speed-testing continued, the requirement was quickly remitted. As the project proceeded, the usage of

subprograms increased. The overhead was minimal compared to the advantages gained in information hiding and reusability.

FIGURE 6 COMPILATION TIME METRICS

COMPILATION TIMES (minutes)

COMPILATION UNIT	LOC	MPX
Null procedure	4	0:17
Null procedure, "with" TEXT_IO	5	0:20
Null procedure, instantiate INT_IO	6	0:24
Null procedure, instantiate FLT_IO	6	0:24
Large compilation, packages	1027	0:36
Math library specification	212	0:29
Math library body	1533	1:09
Math benchmark subroutines	2578	2:03
Complex number specification	61	0:17
Complex number body	167	0:33
LFI package	366	0:31
Varying length string spec	125	0:23
Varying length string body	271	0:48
Varying length string test	62	0:32

FIGURE 7 COMPILATION SIZE METRICS

COMPILATION SIZE BYTES

COMPILATION UNIT	LOC	MPX
Null procedure	4	224
Null procedure, "with" TEXT_IO	5	224
Null procedure, instantiate INT_IO	6	284
Null procedure, instantiate FLT_IO	6	292
Large compilation, packages	1027	204
Math library specification	212	4,536
Math library body	1533	25,544
Math benchmark subroutines	2578	48,640
Complex number specification	61	232
Complex number body	167	5,928
LFI package	366	4,276
Varying length string spec	125	1,236
Varying length string body	271	20,124
Varying length string test	62	6,356

FIGURE 8 EXECUTION SPEED METRICS

EXECUTION SPEED (Microseconds)

OPERATION	Ada MPX	FORTRAN
Internal procedure call (no parameters)	0.7	1.4
Internal procedure call (1 parameter)	1.1	1.7
Internal procedure call (2 parameters)	1.4	2.6
External procedure call (No parameters)	0.7	1.4
Integer assignment	0.3	0.4
Integer addition	0.2	0.2
Integer multiplication	0.2	0.4
Integer division	1.8	2.2
Integer to float conversion	0.3	0.3
Floating point assignment	0.5	0.3
Floating point addition	0.3	0.4
Floating point multiplication	0.7	1.0
Floating point division	1.3	1.6
Exponential function	41.3	14.1
Logarithm function	237.0	15.7
Sine function	49.5	11.0
Tangent function	49.5	14.1
Dynamic allocation, 8 bytes	---	N/A
Dynamic deallocation, 8 bytes	---	N/A
Exception raising/handling	810.0	N/A

N/A: Not Applicable

Figure 9(a) shows metrics gained from designing software in Ada using Boeing methodology. The most important thing to note about these metrics is the time these were produced in the project phase.

FIGURE 9-a SOFTWARE DESIGN METRICS

* LINES OF CODE

FORTRAN	Ada PROJECTION	Ada ACTUAL
665	532	510
	20% REDUCTION	24% REDUCTION

* EXECUTION TIME

FORTRAN	Ada CHECKS ON	Ada CHECKS OFF
2.86 MS	3.62 MS	3.03 MS

These Metrics were gathered at tier level #2 which was design/coded and tested early in the project. Figure 9(b) metrics were gathered on the same data, but later in the project the system designer went back and reimplemented lessons learned to that abstraction. Pragma (In_Line) was not operational for that particular Gould implementation.

FIGURE 9-b SOFTWARE DESIGN METRICS

* LINES OF CODE

FORTRAN	Ada PROJECTION	Ada ACTUAL
665	532	465
	20% REDUCTION	24% REDUCTION

* EXECUTION TIME

FORTRAN	Ada CHECKS ON	Ada CHECKS OFF
2.86ms	3.03ms	2.75ms

Hardware Software Integration (HSI)

During HSI, metrics were gathered over phases of HSI. HSI is broken into two phases - Initial and Fine Tuning. Discrepancy Reports (DRs) are gathered and classified under three main categories:

- Hardware
- Software Requirements
- Software Design

The Hardware DRs consisted of discrepancies in the hardware which impacted HSI, i.e., (broken Airspeed Indicator). NOTE: This is not Ada specific. Software Requirement DRs were written when a requirement for an abstraction was changed, i.e., (Page 42, Ground Flight Status Page 1 of 4, Line 32 should turn red when selected instead of blue; The FAA wind profiles to be simulated are to include five new profiles.) Any items that are software design problems are classified under software design DRs, i.e., (The light illuminates in five seconds instead of six as shown by design criteria). Figure 10 shows the Initial Phase time-line comparisons on the average to solve these types of problems. For fine tuning, the data does not exist yet, but will be available at a later date. The item to note here is the comparison of how the design produced by this method will accept

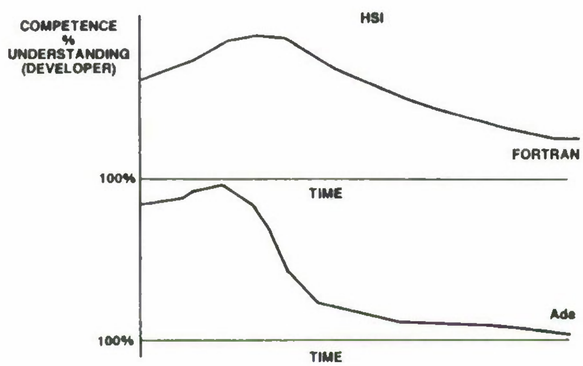
FIGURE 10 HSI PROBLEM SOLVING

TYPE	ANALYSIS	CORRECTION IMPLEMENTED
Requirements	0.5 hrs	2.5 hrs
Hardware	0.1 hrs	Indeterminant
Software	0.11 hrs	1.22 hrs

fine, subjective tuning, and the need/ usage of a real-time monitor.

Figure 11 gives a good comparison of initial assumptions during HSI. The figure clearly shows how the understandability of the language plus the design method chosen leads to shortening the inital phases of HSI.

FIGURE 11 SOFTWARE PROBLEM SOLVING DURING HSI



SUMMARY

This paper has presented an approach to building a real-time system in Ada. The advantages and disadvantages of using the Boeing methodology has been presented. One must realize the difficulty of producing a real-time, maintainable, reusable and loosely-coupled Ada system and plan for the new software revolution today. Hopefully, this paper has exposed some of the knowledge needed to start the transition of industry to software development in Ada. Many projects will be developed before the full understanding of the Ada system and software engineering principles are successfully and fully applied. The use of Ada and its capabilities and attributes will reduce life cycle cost and enhance maintenance. ASVP is just the beginning!

REFERENCES

1. J.G.P. Barnes, Programming in Ada, International Computer Science Series, 1984, Chapters 1-15.
2. J. O'Day, "Design, Development and Management of Reusable Software Components in Ada", Preceedings of 8th Interservice/Industry Training Systems Conference, November 18-20, 1986, Salt Lake City, UT, p. 134-137.
3. ASVP Contract Deliverable Requirements List (CDRL) Documentation. 1986 - 1987.

ABOUT THE AUTHOR

Mr. Jerry Hendrix is the lead technical engineer for the Ada Simulator Validation Program. Mr. Hendrix holds a Bachelor of Aerospace Engineering degree from the Georgia Institute of Technology. Presently, he is pursuing a Master's degree in Industrial Engineering. Mr. Hendrix was trained in Ada at the University of Houston, Clear Lake City, Texas. Prior to joining Boeing, he was employed by Singer Link Flight Simulation Division - Houston.

Experience in Implementing an Ada* Real-Time Program for Flight Simulation Operation

Greco Myren
Honeywell Flight Simulation
13775 McLearan Road
Herndon, Virginia 22071

The use of Ada and reusable software components promises to significantly reduce cost, development time, and improve reliability. This paper reviews the experience gained in implementing an Ada real-time software program (a software crew station for a flight simulator implemented using the Alsys Ada compiler on Sun-3 160M Workstation, in conjunction with a real-time simulation on a Gould 32/8750). First, Ada and the spirit of Ada are briefly reviewed. Then the design and implementation of this Ada program are discussed in terms of design methodologies, design problems, desired speed and time optimization techniques, isolation of machine specific "C" graphic primitives, compiler bugs, debugging problems, efficiency of Ada, and modifiability of existing code. The Ada experience is then related to common areas of concern to industry. Recommendations for Ada development are then given.

Ada and the Spirit of Ada

Ada was not designed to be just another programming language. It was designed to be a programming system that would have features to encourage modern programming practices. Ada has many features such as packages, types, data encapsulation, the Ada Programming Support Environment, and generics, that enable Ada to be a powerful tool to help us understand problems and express solutions in a manner that directly reflects the multidimensional real world. However, Ada can be misused and it is possible to write poor Ada code. Coding in Ada should be approached with the spirit it was designed in: with modern programming practices in mind. This means code should strive to be:

Easily Modifiable	- be able to add changes without increasing the complexity of the original system
Reliable	- operate for long periods of time without human intervention
Understandable	- help us manage the complexity of the software by having a clear design structure at both the high and low levels. The solution should map closely to the real world.
Localized	- all logically related computational resources should be in one module (highly cohesive).
Abstract	- reduce the number of details a programmer needs to know at a given level.
Confirmable	- system should be able to be decomposed so it can be readily tested.
Uniform	- have consistent notation and be free of unnecessary differences.
Modular	- be relatively independent of other modules and have limited interconnection with other modules (loosely coupled)

The Project

The purpose of Honeywell's Ada project was to research what effect Ada would have on flight simulation. Our project included researching design methodologies appropriate to flight simulation, testing out what we considered to be the best design methodology, and writing a Software Crew Station in Ada.

Design Methodology Formulation

Our first task was to evaluate potential modern design methodologies available for Ada software development in flight simulation. We were concerned that the design methodology most appropriate for flight simulation could be different than for the rest of the Ada community. For example, flight simulation is very time critical and must handle hundreds of interrupts each second, and must immediately service each interrupt completely from start to finish. A large part of the Ada community handles only a few interrupts every second and immediate servicing from start to finish is not so important. Hence, if a design methodology produces code that will not execute fast enough on processors suitable for flight simulation, it is of little value.

We identified the potential modern design methodologies to use with Ada and flight simulation and developed a list of desired traits for these methodologies. These methodologies were the Top-Down, Bottom-Up, Data-Flow, Data Structure, Parnas Decomposition, and Object Oriented Design methodologies. The traits used to evaluate the design methodologies are listed in the left hand column in Table A.

Our next task was to categorize the desired traits in order of priority. Several criteria were used in determining the priority of the traits. Most important were execution speed, life-cycle maintainability, and cost. Execution speed is extremely important because if a program can't execute fast enough with 50 percent spare time, and if a program can't be modified to improve the speed in key areas, the program is useless. Life-cycle maintainability is important because engineering change proposals and improvements will be desired later from the program, and the required changes should be possible and easy to make. And cost is important because of the strong push industry wide to reduce software cost.

* Ada is a registered trademark of the U.S. Government, Ada Joint Program Office (AJPO).

Design Methodologies

Desired Traits	Top Down	Bottom Up	Data Flow	Data Structure	Parnas Decompo-sition	Object Oriented Design
First Priority						
Execution Speed	G-E	E	G	M	G	G
Life Cycle Maint.	G	M	M	M	E	E
Program Reliability	M	M	G	M	G	G
Debugging Ease	M	M	M	M	E	E
Low Paperwork	M-G	M	M	M	G	G
Appropriateness to Ada	E	G	M	M	E	E
Second Priority:						
Large System Mgmt	M	B	G	M	G	G-E
Incorporating Changes	M	M	M	G	G	G
Uncovering Inconsitencies Early	G	B	M	M	G	G
Modules Loosely Coupled	M	M	B	B	G	G
Modules Highly Cohesive	M	M	M	M	M	E
Help Appropriately Name Program Elements	M	M	M	M	M	E
Third Priority:						
Understandability	M	B-M	M	M	G	G
Designer Productivity	G	M	M	M	G	G
Strongly Defined Guidelines	M	M	G	G	M	M
Evenness of Machine Time Demand	G	B	M	M	G	G
Low Number of Test Drivers	G	B	M	M	G	G
Fourth Priority:						
Evenly Sized Modules	M	M	B	M	G	G
Minimum Dependency on Designer Experience	M	M	G	G	M	M

Lengend: E=Excellent G=Good M=Medium B=Bad

Table A. Design Methodologies and their Appropriateness to Flight Simulation Desired Traits when using Ada.

The next criterion used in prioritizing the traits was the production of quality software. We felt this to be very important, but did not give it as much emphasis as the other criteria because we felt that Ada itself will help significantly produce quality software.

The remaining criteria used in prioritizing the traits were the appropriateness of the design methodology trait to configuration management, to human understanding, to Ada and to large system design.

The resulting flight simulation priority groupings are shown in Table A along with how they rated with different design methodologies. This rating was all done with flight simulation in mind and may differ outside the flight simulation environment.

Top Down, Parnas Decompoeition, and Object Oriented Design became the front running design methodologies as a result of their rating in the desired traits of the first priority. All three design methodologies rated high with the appropriateness to Ada. Top-Down ranked best in execution speed, while Parnas Decomposition and OOD rated best when it came to life cycle maintainability, debugging ease, and program reliability.

In a comparison of these three methodologies with the desired traits of the second priority, Parnae Decompoeition and OOD stood out. OOD rated high when it came to large system management and production of highly cohesive modules. A very interesting and appealing trait of OOD was how high it rated in helping name program elements appropriately.

The desired traits of the third priority showed Top-Down, Bottom-Up, and OOD as equal when it came to design production, strongly defined guidelines, evenness of machine time demand, and low number of test drivers. However, Top-Down did not rank as well as the other two design methodologies when it came to understandability.

In desired traits of the fourth priority, Parnas Decomposition and OOD were rated better on even-size module production, but all three design methodologies rated low on minimum dependency on programmer design experience.

The results of this first study indicated that the beet potential design methodologies were the OOD, Parnas Decomposition, and Top-Down design methodologies. OOD appeared to be the most appropriate design methodology of the three for flight simulation. OOD stood out in the desired flight simulation methodology traits of life cycle

maintainability, program reliability, debugging ease, large system management, incorporation of changes, production of highly cohesive and loosely coupled modules, help in appropriately naming program elements, understandability, and the production of evenly sized modules. However, OOD execution speed did not rate as high as that of the Top-Down and Bottom-Up design methodologies. It was believed at this point that OOD would be fast enough for flight simulation purposes; so we chose to continue the study of OOD.

Object Oriented Design Case Study

The purpose of this study was to test OOD on a typical flight simulation module. We defined a typical flight simulation module as one that runs at 10 Hz, gets data from hardware, contains a fair amount of logic and reasoning, has equation processing, and is dependent on variables from datapool (global data). Out of the many modules that meet these criteria, the landing gear module was selected as it was a module that most people intuitively understand.

The OOD methodology we chose to follow was the method offered in Grady Booch's Software Engineering with Ada as it is a very widely used book in the Ada community. EVB Software Engineering's handbook An Object Oriented Design was also used, as it expands on Booch's method of OOD.

We followed all the detailed OOD steps in our study. The informal paragraph was written (figure 1.), nouns and adjectives underlined, object and identifiers determined, a list of operations on the object formulated (figure 2.), interfaces established, Booch diagrams drawn (figure 3.), package specification written, package bodies written, and tasks, procedures, and functions coded. A large number of Ada tasks were appearing for two reasons. First, tasks were necessary to handle the hardware input from the buttons the pilot selected in the cockpit. The number of tasks necessary to handle button input varied from one to three, depending on how the designer chose to divvy the objects up. And secondly, tasks were necessary to raise and lower the wheels. A task would have to be activated 10 times a second for at least eight continuous seconds to allow the wheels to be raised or lowered a certain delta each time. This study yielded some interesting results, which are listed in table B.

The pilot may chose to raise or lower the landing gear by pushing the UP or DOWN button. There is a safety lock on the Up button that prevents it from being pushed if the aircraft is on the ground. This safety lock is activated by the "weight on wheels" microswitch on each wheel. If the pilot wishes to transfer the landing gear control to the rear cockpit, he may do so by pressing the transfer button. By pressing the transfer button again, control returns to the front cockpit.

There is a standby lowering and a standby raising landing gear system which can be used in case the landing gear fails to lower or raise normally. The standby landing gear raising system can be activated by turning the UP button clockwise 60 degrees and then pressing it. The standby lowering system is activated by pulling the UT/C handle. When either of the standby systems is activated, the normal landing gear selection system is de-energized. Either system can be activated only once, and as a precaution, the standby raising system cannot be activated if the standby lowering system has been activated previously.

Figure 1. Informal Paragraph For the Landing Gear of a T-45 Aircraft with the Verbs and Adjectives Underlined (Operations on the Objects).

```
[ ] Normal_Landing_Gear_Controls
        Front_Up_Button_Enabled
        Rear_Up_Button_Enabled
        Front_Down_Button_Enabled
        Rear_Down_Button_Enabled
        Front_Transfer_Button_Enabled
        Rear_Transfer_Button_Enabled
        Weight_On_Wheels_Button_Enabled
        Deenergize

[ ] Standby_Lower_Landing_Gear_Controls
        Front_UT/C_Handle_Pulled      (Front_Standby_Lower_Enabled)
        Rear_Turn_&_Push_Up_Button    (Rear_Standby_Lower_Enabled)
        Deenergize

[ ] Standby_Raise_Landing_Gear_Controls
        Front_Turn_&_Push_Up_Button   (Front_Standby_Raise_Enabled)
        Rear_Turn_&_Push_Up_Button    (Rear_Standby_Raise_Enabled)
        Deenergize

[ ] Landing_Gear
        Raise
        Lower

[ ] Weight_On_Wheels_Controls
        Left_Wheel_Activate
        Left_Wheel_Deactivate
        Right_Wheel_Activate
        Left_Wheel_Deactivate
        Is_Activated
```

Figure 2. The Resulting List of the Operations on the Landing Gear.

Positive Results:

1) Modules are loosely coupled and highly cohesive
2) Resulting code is very readable
3) Software solution maps closely to the real world
4) Design decisions are localized
5) Strict definition of object interfaces
6) Blends well with Ada
7) Code evolves quickly from design
8) Focuses on objects and operations like the real world

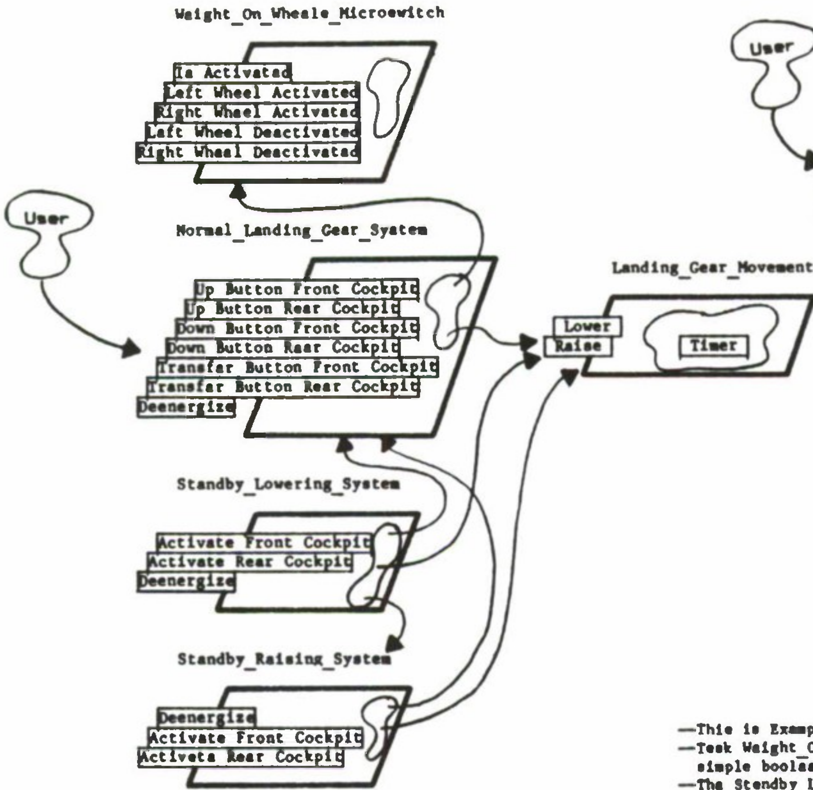
Negative Results:

1) Takes time for designer to become adjusted to concepts, techniques, and guidelines
2) Lacks mechanisms for determining requirements. Some other design methodology will be required as an initial effort of a very large software development effort.
3) Intuitive choices must be made by the designer. Two different programmers given the same problem may come up with two different solutions.
4) OOD's informal paragraph is hard to write. What looks good may turn out later to need rewriting.
5) OOD appears to be at an infant stage.
   - design steps need to be defined better
   - many items need to be expanded upon
   - more feedback from private industry is needed to better define OOD
   - more and better example are needed; in the real world tougher problems exist than those presented.
6) OOD as presently defined will cause flight simulation to rely heavily on Ada tasking.
   - for a large system too many tasks may be produced.

Overall Conclusions:

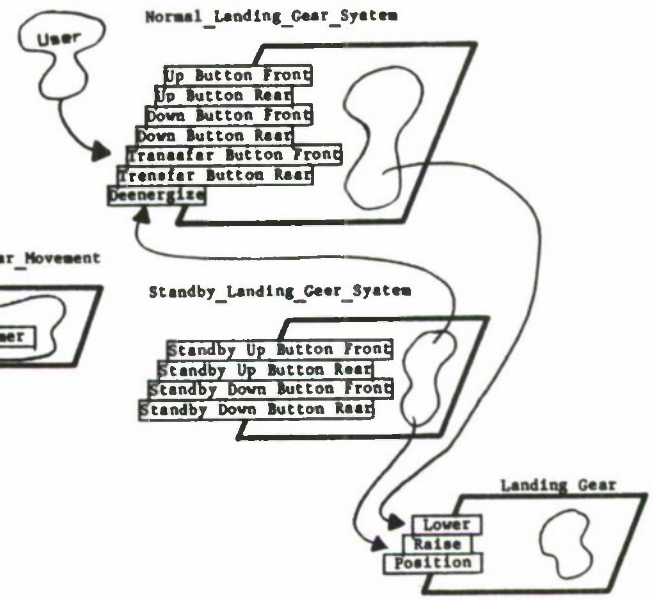
1) The resulting Ada code is very readable and intuitively understandable.
2) OOD looks good for flight simulation
3) OOD instructions and guidelines need to be improved
4) OOD is medium in difficulty to learn and use, but can be taught to anyone with programming experience.
5) for large projects some other methodology is recommended to determine requirements, and partition the problem.
6) OOD can be modified, so that it does not depend on Ada

Table B. Results from the Object Oriented Design Case Study



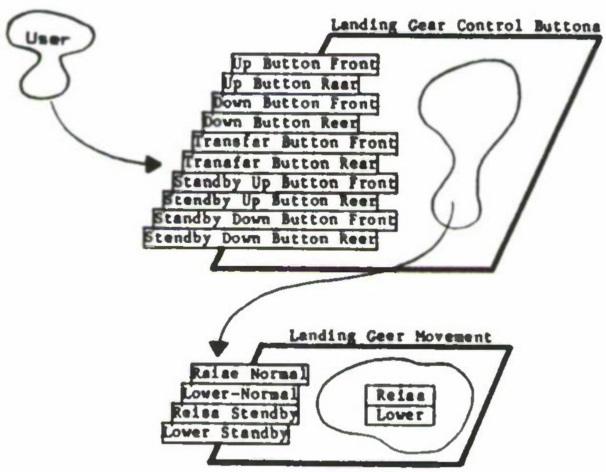
Booch Diagram for Example 1

- Note that five different Ada tasks are being used.
- How much actual code each task will contain will not be known until coding later on.
- Five tasks is too many tasks for this module. If the rest of the system used this many tasks, the simulator would have 400-1000 tasks which would create too much context switching overhead, resulting in a slow system response.



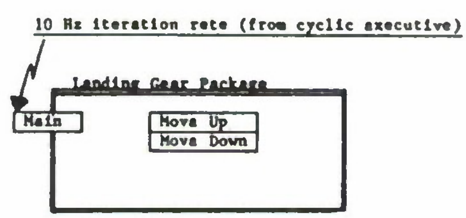
Booch Diagram for Example 2

—This is Example 1 redone into 3 tasks.
—Task Weight_On_Wheels_Microswitch for example 1 is assumed to be a simple boolean flag and not included here.
—The Standby_Lowering_System task and the Standby_Raising_System task have been combined into the Standby_Landing_Gear_System task.
—The Normal_Landing_Gear_System task and the Standby_Landing_Gear_System task will only be activated when a button is pressed. Otherwise, they will be passive.
—The Landing_Gear task is activated by a lower or a raise command. It will stay active (awaking itself every cycle) to complete the lowering or raising of the landing gear.



Example 3

—This is Example 1 combined into only 2 tasks.
—All landing gear control inputs are handled in 1 task.



Example 4

Notes:

— There are no Ada tasks (no rendezvous with other tasks either).
— There are no built-in clocks ("delays").
— A cyclic executive is assumed to use this package at 10 Hz.
— The package is activated simply by calling procedure Main.

Figure 3. The Four Object Oriented Design Solutions for the Landing Gear.

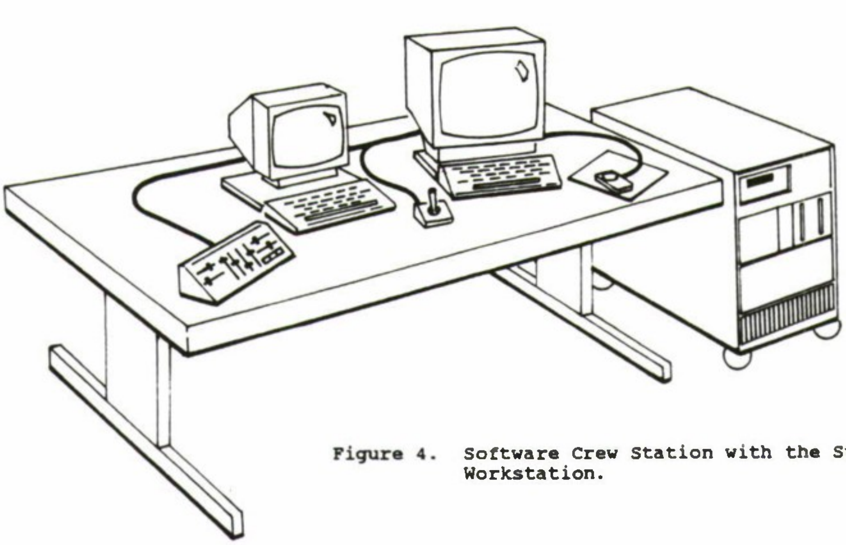


Figure 4. Software Crew Station with the Sun-3 Workstation.

The Software Crew Station

Our next task was to gain Ada experience by writing actual Ada code. Since the Ada compiler for the SEL computer was not available at the time, we purchased a SUN-3 160M workstation for which a validated Ada compiler, the Alsys Ada compiler (version 1.1 for the SUN), was available. We wanted to write a program that could be of use to the company. We elected to write a software crew station in Ada, which could be used to test flight simulation software. (The purpose of a software crew station is to test in real time for software errors in modules that have completed independent testing, but have not yet entered the hardware software integration (HSI) phase).

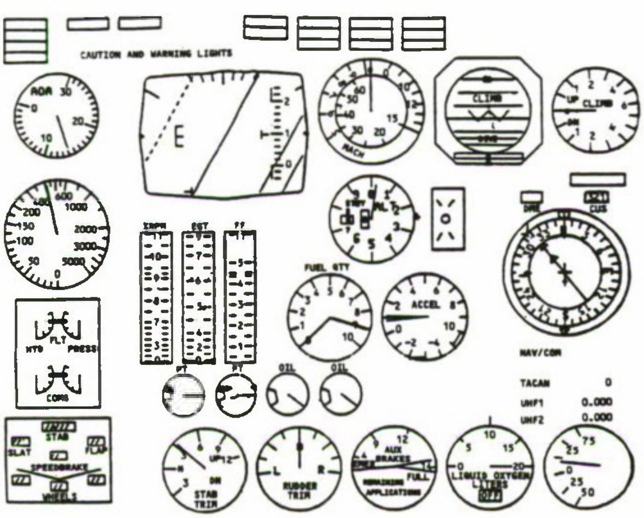


Figure 5. An Example Software Cockpit Display.

Speed Considerations

Before we designed the software crew station, we were aware that we might have speed problems in three areas:

1) The speed over Ethernet from the joystick on a PC to the SEL 32/9750 and then to the SUN-3 Workstation (where it was up to Unix to handle the input).
2) The speed of the different graphic packages available on the SUN.
3) The speed of Ada.

We figured that if there were going to be speed problems with Ethernet, there probably wasn't too much we could do, and we chose to ignore it as our main purpose was to get experience in Ada. When it came to the speed of Ada, we decided not to stray too much from good design for time optimization; so again we chose not worry about it. But when it came to the different graphic packages available on the SUN, we decided that there was things we could do. We researched the graphic packages available to us, which were SUNCORE, SUN CGI, and PIXRECT (no GKS). What we decided to do to increase speed was the following:

1) Use 2-D graphics instead of 3-D (speed increase of a factor of 2-4)
2) Use the lowest level of graphics, Pixrect, whenever possible for the fastest speed (Pixrect basically offers lines and text).
3) Use the next highest level graphics, SUN CGI, only when circles, arcs, and mouse control were required.
4) Not use the highest level graphics, SUNCORE, which is 7-25 times slower (but offers windows, sliders and scrollbars).

Because we were giving up some high level graphic features we desired (windows, scrollbars, etc.), and because we were going to be using two different graphics packages, Pixrect and SUN CGI, we were going to have to:

1) Do all the bookkeeping in Ada (coordinates, scalings).
2) Make our own graphic commands in Ada (boxes, dials, needles).
3) Write Ada to C interfaces for the calls to the Pixrect and SUN CGI graphics.

We felt this would be a interesting program because it had the potential of getting very complex.

Design Problems

The Software Crew Station was designed using Object Oriented Design, but without writing the informal paragraph. The design was fairly easy because this application blended well with OOD and because the informal paragraph was not written.

One of the interesting design problems was in using Ada Generics. Ada generics are easy to use and so we designed a generic instrument using generics. We kept getting more and more generic, adding features such as being able to move the instrument anywhere on the screen, go clockwise or

counter clockwise, use real or integer input, use different types of labeling, or being able to enlarge or shrink the dial. But when we instantiated 18 dials, we overloaded the name space on the compiler. What happened is that the generic dial package we created had about 30 procedures and functions, and each time we created an instrument we would create 30 new and unique procedure and functions. In essence we had 30 * 18 = 540 name units for the dials alone.

Our conclusions are:

1) Ada generics are easy to write.
2) You can get carried away while making something generic. If it is made too generic, it becomes increasingly hard to read and intuitively understand.
3) You can quickly use up the name space on Ada compilers.

Isolation of Machine-Specific "C" Graphic Primitives

First, we had to determine which PIXRECT and SUN CGI graphic commands were needed. Then we had to write Ada to "C" interfaces for each graphic command. All the interfaces for the PIXRECT graphic commands were placed in an Ada package we called PIXRECT, and the same was done for the SUN CGI graphic commands. On top of those two packages we made a package called Ada_Graphics, which would be the graphic package our Ada Software Crew Station would use. Ada_Graphics would know which graphics (Pixrect or SUN CGI) to use to draw lines, circles, text, and would keep track of the coordinate system for each. Also Ada_Graphics would contain graphic functions that we would create such as Draw_Box and Shrink.

Writing the interface package from Ada to C was fairly simple and straightforward, following the guidelines for interfacing from Alsys. Writing a driver to test out the package was also simple, as all we had to do was "with" the package. Writing the actual "C" routines was not so simple. Here the difference between Ada and "C"became evident: Ada code is fairly straightforward while "C" is not.

One problem we encountered when interfacing Ada to C was how to write an Ada program that has pointers to C structures. Since the Ada program was going to do all the bookkeeping, it was going to have to pass the pointers to C structures that each Pixrect and SUN CGI graphic command requires to operate. The solution was to start off having Ada point (using the Ada access structure) to a bogus record, call a "C" routine which would clobber the pointer and make it point to the proper "C" structure. From then on the Ada pointers would be pointing at the proper "C" structure the "C" routine needed to operate. Some examples of "C" structures that were required were the screen structure, device class structure, current font, and view description structure.

Desired Speed and Time Optimization Techniques

One of the first problems we encountered with Ada was when compiling packages using the Math_Lib. The Math_Lib package is generic and to use it you must instantiate it with the desired number of digits. This is a nice feature, but became a nuisance because every time a program was compiled, it would get take about 12 minutes each time to compile the statement:

```
package My_Math_Lib is new Math_Lib(digits=>5);
```

It would take that long because Ada would instantiate every math function call available such as Sin, Cos, *, +, Mod. We knew that we would always want a precision of digits=>5 so we created a "skin" package. All a skin package does is instantiate the precision of digits. A skin package looks like this:

```
with System; use System;
with Math_Lib;
package Skin_Math_Lib is
   type My_Real is digits 5;
   package My_Math_Lib is new
      Math_Lib(real=>Skin_Math_Lib.My_Real);
   use My_Math_Lib;
   subtype Real is My_Real;
   function Sin(X:real) return real   renames
      My_Math_Lib.Sin;
   function Cos(X:real) return real   renames
      My_math_Lib.Cos;
end Skin_Math_Lib;
```

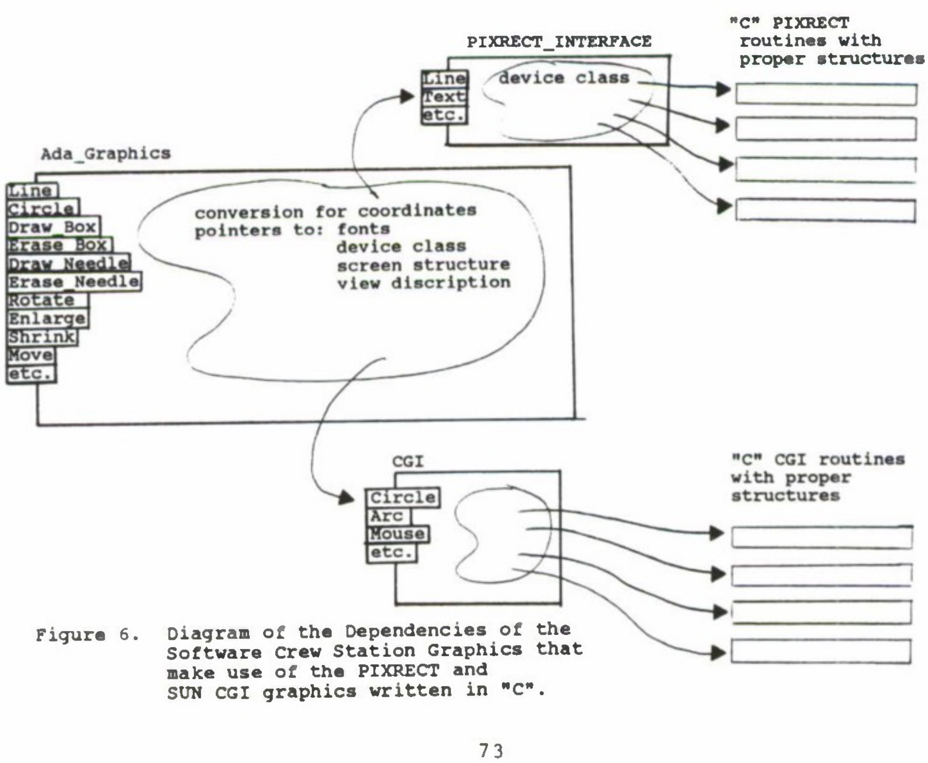


Figure 6. Diagram of the Dependencies of the Software Crew Station Graphics that make use of the PIXRECT and SUN CGI graphics written in "C".

Now every time a package was compiled, we would "with" Skin_Math_Lib and it would take less than a second to have the Math_Lib functions we wanted available.

Another item to speed compilation up for a large program was to use the "is separate" clause. This breaks the program into smaller pieces. When debugging a routine and constantly making changes to one of these pieces, only that piece has to be recompiled.

Compiler bugs

There weren't any major compiler bugs. The compiler flagged all kinds of errors on our part. The messages weren't always clear or concise, but that was not so important since flagging the line the compilation error was at, we usually knew what the error was without the error message.

One bug in the Alsys Ada compiler was that you could not compile a package specification without the body. The remedy to this was to attach at a null body:

```
package body Package_Name is
begin
   null;
end Package_Name;
```

A second problem with the Alsys compiler(version 1.1) was with using the floating point processor: it compiler did not use it. Alsys did all the work in software. Needless to say this was very slow.

When we did get the new Alsys compiler that would make use of the floating point processor, it wouldn't work on the SUN workstation because it also required the newest version of the UNIX operating system. So we had to go order and wait for the latest Unix operating system on the SUN.

Compilation Environment

Once the program got large enough we found that many times it was necessary to go back and change or add some variable. This would necessitate recompiling many package specs, body, or stubs that were effected. We needed a chart to show all the items that needed to be recompiled and in which order. Then each spec, body, and stub had to be compiled separately. This got to be a nuisance and a waste of programmers time to sit at the terminal for 20 minutes compiling all the modules. It would have been nice to have an Ada environments which knew which modules had been modified and in turn which modules needed to be recompiled by order dependencies. One of these environments will probably be necessary for a large Ada system.

Efficiency of Ada

Since we do not have a comparable Fortran program to use as a benchmark, we cannot compare the resulting speed. However several noteworthy items should be pointed out.

The resulting code is very modular. This is a very key concept. Each package is very easy to test. It is easy to make changes to improve speed in any routine and not worry about what effects they are going to have on the rest of the system. A new application program can use existing packages directly without making any changes to the package whatsoever.

The resulting code is very readable. Someone years later will not hesitate to use the package because the way it works is very straightforward. There are no hidden surprises.

Our conclusion is that Ada is ahead the game. If anything, Ada is waiting for the speed of hardware to catch up.

Common Areas of Concern to Industry

Based on our experience, this is how we would answer these common areas of concern to industry:

Is Ada too complex? No, not when you use it in the context it was designed for; modern design methodologies. In fact, Ada helps you organize and keep things simple.

Are Ada compilers slow? Yes, and probably always will be. Ada compilers are getting optimized, but since they do so much more error checking than Fortran or Pascal, it means Ada compilers must do a great deal more bookkeeping. This will always be the case; so Ada compilers will probably always be slower than Fortran or Pascal compilers. However, Ada catches more errors at compile time which is a lot faster, easier, and less costly than having to find them later on.

Is Ada hard to learn? No, not for a programmer who is familiar with modern programming practices. It is hard for someone who isn't, not because the Ada syntax is difficult to learn, but because Ada should be used with modern programming practices. The big learning curves are these practices.

Are Ada PDL's worthwhile? Yes. Especially Ada based PDL's that require defining objects (data and types). Ada PDL's can be a good tool to teach Ada to students.

Will Ada reduce development time? It depends. When companies start out using Ada they will find that the design will take longer, but the integration and testing will be shorter. So, on the first Ada attempts it will probably take overall the same as it always has. However, on the subsequent Ada projects development time will be reduced because much existing Ada code will be able be reused in its entirety or easily modifiable, much more than Fortran was ever able to.

Will Ada reduce cost? Yes, of course in the long run, for the government. In the short run, for new programs, probably not. However, as mentioned above, as the years go by, companies will bid lower costs because they will be able to reuse more and more software.

Is Object Oriented Design the way to go with Ada? Yes. One of the strong points of OOD is its ability to properly name objects and operations. The resulting code is so readable and is so much as we humans think, that one hardly needs to read the documentation or PDL.

Is Ada tasking slow? Yes. The industry is doing a lot to help out in this area because, after all, this is what Ada was designed for. But what many people don't understand is that you don't have to use Ada tasking in order to use Ada. You can use the conventional cyclic executive with Ada code, and many times it is necessary.

Is Ada execution speed slow? Yes. Usually about 30 percent slower. The use of Ada constants help improve speed. Theoretically, Ada can be made to be faster than Fortran, but this will take a sophisticated compiler. However, speed of hardware is always being made faster each year. The benefits of Ada outweigh it's slowness. Many people get excited about Ada's slowness in light of the requirement for 50 percent spare time and in light of programs becoming more sophisticated. However, good design should not be bent too far to accommodate speed. More computer power is one solution. Interface to assembler is another great solution for bottlenecks. A program does not have to be 100 percent Ada.

Is Ada truly transportable? Theoretically, yes, Realistically, no. There are too many items that will depend on each machine. However, most of these machine dependent features can now be localized. Transporting programs to another machine will be much, much easier than it has ever been.

Will Ada really make it? Yes, emphatically yes. There are just too many benefits to stop it. The fact that it is 30 percent slower than Fortran is a bad reason not to use Ada for all its benefits. With the increase of speed of hardware, this excuse will go away. Industry has been hesitant to the transition to Ada. Industry is comfortable with Fortran and other languages that have been around for a while. While these languages can't match Ada, industry has mastered these languages and understands them quite well in spite of their quirks. Industry has so much invested in these languages, why risk converting over to Ada, an area which appears full of uncertainties, which they aren't experts in, and which none of their employees know. Fortunately, the Government now requires Ada, and the transition will occur. Would the change have occurred without government intervention? It is very doubtful.

Recommendation for Ada Development

From our experience, we have to following to recommend.

1) Ada Code is made very readable via the use of Object Oriented Design. We recommend using OOD. Beware that, OOD is not well defined. We recommend not using the informal paragraph as it appears to make OOD more difficult to learn.
2) Look into Ada environments that can handle compilation dependencies.
3) Beware of overusing Ada Generics. While generics is easy to use, don't try to make everything generic. Generics can create an enormous amount of overhead. Also, if a program is made too generic, it becomes difficult understand and use.
4) Ada compilers are slow. Look into distributing the compilation demands among different machines for program development. Also, when a system nears completion and recompilation of an entire system necessary, this has the potential of taking several hours for a large system.
5) Managing Ada projects will become easier for managers. Managers should not be afraid, although they may not have coded for years. Ada code will be easier to understand, where a program fits in the entire system will be more readily apparent, and the impacts a change will have on the entire system will be known.
6) Testing Ada code is easy. Go ahead and write a driver that "withs" your program and tests out your procedures.
7) Ada will be hard to learn for programmers who have no experience with modern design methodologies. Be prepared to teach modern design methodologies along with Ada.
8) Use Booch-like diagrams to explain your Ada system to others. They offer a quick graphical way to convey your system structure.

About the Author

Mr. Myren has been doing Ada research for the last two years for Honeywell Flight Simulation. He has taught Ada courses at Unisys and Honeywell, and has written a 236 page Ada Tutorial and Study Guide. He has worked on CATCC_DAIR (Navy's carrier automatic air traffic control system), OUTBOARD (Navy's communication between ships), and has 5 years of software experience with compilers, operating systems, and real time software. He holds a BS in Computer Science from North Carolina State University. Mr. Myren is currently employed at Quintron Coorporation, Chantilly, Virginia.

ADA® IMPLEMENTATION IN MULTI-DEVICE CONFIGURATION

S. Ramachandran
D. McCabe

McDonnell Douglas Helicopter Company
Mesa, Arizona

ABSTRACT

This paper examines Ada implementation of a multi-device configuration in an engineering organization. The advantages and disadvantages of Ada are examined from this perspective. System architecture, software development environment, Ada compilers/cross-compilers and software development environment, Ada compilers/cross-compilers and software engineering methodologies are discussed. Simulation architecture selected by McDonnell Douglas Helicopter Company and lessons learned are presented.

INTRODUCTION

Ada has been mandated as the primary programming language by the Department of Defense (DoD) for mission critical programs. This emphasis on Ada has been reflected in recent training systems initiatives also. Since new training devices are built from the ground up, design and development of new simulators are somewhat straightforward. However, aircraft companies with established simulation facilities face the difficult choice of whether and how to make the transition from a mostly assembler/FORTRAN environment to Ada. McDonnell Douglas Helicopter Company is one of the companies that has been making this difficult and expensive transition. This paper examines the technical issues that are to be considered and resolved to make a successful transition. The discussion here is from the viewpoint of an organization that has to support aircraft development with multi-ship simulation. The major question is should the simulation organization switch over to Ada at all or maintain status quo to the software development? This requires an examination of not only the technical pros and cons of Ada software development but also of the charter for the simulation organization (engineering simulation versus training simulation) and the costs associated with the decision to be taken. The paper first discusses the pros and cons of choosing or not choosing Ada. Problems of transition in the light of various stages of development of different simulators are discussed. Once the decision to choose Ada has been made, the technical issues for a successful simulation implementation are reviewed. McDonnell Douglas Helicopter Company's approach to satisfy its simulation requirements is presented. The lessons learned in achieving Ada implementation are also presented.

ADVANTAGES AND DISADVANTAGES OF ADA

The advantages of using Ada in simulation applications are not much different from those for other applications. Engineering simulation is a large software intensive activity. However, in the context of aircraft development, simulation is a medium where the technical efforts of diverse engineering organizations within the company are brought to focus and where airplane concepts are validated. It is also a tool where software eventually intended for aircraft is developed and tested. For these reasons, the advantages of Ada are even more attractive for simulation application. One such consideration is the easier portability Ada offers. Portability of code has been an elusive goal of simulation software as it is with other software applications. It has been recognized for some time that standardizing on a single language will be a major part of making this goal a reality. At this time the Ada language is being far more tightly controlled than any other language in both the language revision aspect (there is only <u>ONE</u> version of the language) and the compiler implementation aspect (certified validation suites). This makes Ada a stable language to standardize on, making portable code far more feasible than with a non-regulated language. Since aircraft development has become software intensive, it is extremely important to reduce software costs. To achieve this, software must be portable between the simulator, hot bench and aircraft. With Ada mandated as the higher order programming language for aircraft development, adopting Ada as the programming language for the simulator also makes good economical sense as well.

Another advantage is that, in the long term, all programmers will use a common programming language and therefore will have a far easier time transitioning from one aircraft project to another. Ada will also permit efficient utilization of programmers since they could be moved around within the company between simulation and aircraft programs depending upon company needs.

The technical advantages of Ada are even more alluring in the context of commonality between simulators, hotbenches and aircraft. Ada encourages structured object oriented design, which closely resembles the way different systems/subsystems/components operate in an aircraft. Built into the Ada language is the construct of packages which allows a

®Ada is a registered trademark of DoD Ada Joint Program Office.

mechanism for putting all the code which describes an object and its processes into a logical unit. This package can be incorporated with other packages or subprograms thereby allowing the use of these code objects throughout a software system. It is a powerful way of letting the code reflect the objects and processes necessary to control a system in an understandable manner. Ada enforces a high degree of structure by imposing the principles of modularity, abstraction, information hiding, and localization. The interfaces are embodied in the package specification and can be totally defined prior to having to work out the algorithms associated with them. Ada decreases the possibility of having the wrong variables being passed from one software unit to another thereby increasing the understanding of the flow of the software and making maintainability and modification easier.

Ada permits depiction of parallel events in an understandable manner. Most languages address this problem by interfacing from their high level language to either operating system calls (which vary from operating system to operating system) or to assembler routines which schedule multiple programs simultaneously. In Ada this concept is addressed right in the language via the TASK construct. The scheduling of simultaneous events is no longer buried in code as a call to some routine which is written in a low-level language and one has to guess what it is doing. In Ada it is labeled as a TASK with clear rendezvous points.It is in the same language as the rest of the code. This is another invaluable advantage for understanding the code for either maintenance or modification purposes as well as emulating the aircraft hardware operation in the simulator.

However, these advantages have to be weighed against the disadvantages of using Ada. One problem area is the lack of compilers with the system dependent features described in the Ada Language Reference Manual's (LRM) Chapter 13. These features include many of the system programming capabilities which are necessary for simulation applications. These include pragmas (which provide the selection criteria for mapping an entity onto the underlying machine) such as PACK (elimination of gaps in storage areas allocated to consecutive components), INLINE (machine code insertion), and INTERFACE (calling subprograms written in another language). They also include REPRESENTATION CLAUSES (imposing certain characteristics of the mapping of an entity onto the underlying machine), ADDRESS CLAUSES (specification of a required address in storage which allow interrupts to be coded), UNCHECKED DEALLOCATION and UNCHECKED CONVERSION. Although there are plans to include these features in future validation efforts, they are not tested at this time. Therefore, one of the important criteria in considering an Ada compiler is what aspects of Chapter 13 are implemented and to what degree.

Another disadvantage with the implementation of the language is the lack of optimization in both host and target Ada compilers and cross-compilers. There are at least two reasons for this. One is that Ada embodies many aspects of computing that were previously rendered to the realm of operating systems. There is a learning curve involved in just being able to implement these aspects in a high order language. The other is that Ada is relatively new compared to other high order languages. Enough time has not passed for optimization to have been the prime focus of the implementors.

Another disadvantage is the conspicuous lack of software engineers capable of designing software which incorporates the unique features of Ada. A structured methodology of design is mandated by these features and changing the way software is normally developed in a simulation environment can be painful, especially if there are not enough experienced personnel to guide the inexperienced programmers. This disadvantage can be alleviated if concerted efforts are introduced to train programmers in the design aspects necessary to properly use this language.

The above disadvantages will diminish in impact if not totally disappear as the implementations of the language mature and the software engineering community gains experience with it.

Despite the implementation shortcomings of Ada, there are two important considerations for adopting Ada. One is whether the simulation facility is interested in training device contracts. Based on current trends in military training system procurement, it is obvious that if a switch is not made to Ada, the company will be left behind in the training market since more and more military training device contracts require Ada and the company will not have the experience or talent base to compete. If the simulation department does not choose Ada as its standard software development language, it will be required to change existing developed code for every new version of its current standard language, whatever it may be. This is due to a lack of tight control on other languages which has been imposed on the Ada language and implemented through the validation process.

However, the transition to Ada is an expensive proposition since most of the existing software is in a language other than Ada, and usually in FORTRAN. This existing code may have to be redesigned in Ada. This cost could be very high since a software redesign rather than a simple conversion is necessary to take advantage of Ada's strengths. New computers, operating systems, and compilers may have to be bought for the implementation since existing computing systems in many cases may not have Ada development and real-time execution capability. Some existing configurations permit Ada-only

implementations and not simultaneous implementations of new Ada code with existing FORTRAN software.

Putting off the switch to Ada will only make it even more expensive later since additional software will need to be eventually redone in Ada. It is also important to maintain a smooth transition to an Ada environment. Ada software conversion/development has to be achieved with an existing workforce and in an economical fashion while supporting the current simulator operation, developing software for new simulators and planning for future ones.

SIMULATION REQUIREMENTS

McDonnell Douglas Helicopter Company's Engineering and Training Simulation Department (ETSD) was faced with the issue of making a decision regarding Ada in early 1986. At that time ETSD had a full mission simulator in operation in support of a research rotorcraft program. At the same time the department was in the early stages of developing a second simulator for an advanced version of an existing rotorcraft. Plans were also being made for developing a third simulator in 1987 for this program. A major part of the simulation software on the first simulator was in FORTRAN with the rest in assembler and C.

All the simulated aircraft incorporate the latest and planned advances in combat rotorcraft technology including "glass cockpits" and a full suite of avionics, weapons, and sensors. The simulators provide full flight and mission simulation including visual and sensor simulation along with moving map systems. Since the first simulator was developed during 1984, Ada was not used. Consequently, the issue of Ada was taken up later when the second simulator program was started.

Along with the new simulators McDonnell Douglas Helicopter Company also faced the issue of interactive simulation. The aircraft could participate in joint missions in mixed roles of adversaries or friendlies. The development of a system control station for typical instructor/operator functions as well as for engineers to obtain and analyze aircraft data also required the networking of these three simulators plus other simulators as they were developed and brought on line. The requirement for the second and third simulators gave the opportunity to examine technological alternatives in terms of both hardware and software in the light of recent computer technology and Ada implementation.

TECHNICAL ISSUES

The technical issues to be resolved included computer hardware performance and requirements,software development environment, software tools, Ada compilers/cross-compilers, and software engineering.

System architecture:

One of the guidelines used in the hardware evaluation was the need to use existing minicomputers as well as a special purpose processor for high speed flight dynamics and control simulation. A modular design with minimum system upgrade costs was desired since the demand for computer power invariably increases with enhancements to aircraft capability. To facilitate effective man-machine interface between the pilot and the helicopter, minimum data transport delay between processors is required. The configuration must support implementation of a wide variety of algorithms ranging from artificial intelligence to aircraft electrical and hydraulic system. The system configuration must also permit simulation at rates up to 60 Hz. Most importantly, the hardware cost must be as low as possible. The popular criterion of cost per MIPS was used as a yardstick to determine hardware cost. A wide variety of systems were evaluated including multiprocessors, multicomputers, array processors and pipeline systems. Different vendor products within each group were also evaluated. Based on the above requirements and the implementation of Ada, it was found that distributed processing with multi-processors offered the best cost and technical solution.

Software Development Environment:

While the above resolves the target or the real-time implementation system, it is equally important to address the issue of the software development system for simulation. Developing software in Ada is different from developing software in other languages due to the many aspects of this language discussed earlier. Due to this difference, the development environment becomes an extremely important tool. The necessity to understand how different types of software modules interact with each other in a multi-tasking and generic software system drives the need for software tools which allow graphical/textual representations of design and automatic documentation, as well as appropriate compilers and cross-compilers.

During the preliminary design phase of development automatic software tools for graphical representations of data flows and module constructs such as packages, tasks, and subprograms allow the software designers a standardized way of creating their designs and an excellent method for documenting it in an understandable form. During the detailed design phase, textual representation of the algorithms and variables is more uniformly presented with the aid of Programming Design Languages (PDL) and Data Dictionaries. Many of the commercial PDLs on the market also include templates which produce them in military standard formats as well as supplying automatic metrics. As pointed out earlier, the implementors of the Ada language have been slow to implement all the

features in the Language Reference Manual. It therefore becomes critical to develop the criteria for evaluating compilers and cross-compilers for simulation applications in order to avoid problems at the coding phase. Appendix F of every compiler's manual and Chapter 13 of the Language Reference Manual provide the real-time features which may be necessary for simulation applications.

Once it is realized that software tools become more of a necessity when designing in this language, the question of whether the present development system is adequate becomes very important. It requires analyzing the existing development system, software tool requirements, software development tools that are currently available, and most importantly, if they will work together. If the tools do not work together, then the question is whether the present development system for Ada design and coding should be used.

Ada compilers/cross-compilers:

To be cost effective the real-time (target) system may be quite different from that of the software development system, as was the case at McDonnell Douglas Helicopter Company. In this case it is important to evaluate not only the compilers but also the cross-compilers. These are the most important software tools in the development environment. Designing a list of criteria for necessary and desirable features in an Ada compiler/cross-compiler for a given application can make the job of selecting the compiler/cross-compiler easier. Simulation code is run in real-time and therefore criteria such as ADDRESS CLAUSES (to allow interrupt capability) and pragma INTERFACE to the target systems assembly language may be necessary. The pragma PACK and REPRESENTATION CLAUSES may be added to the list of desirable compiler criteria if transporting of large volumes of data throughout the simulation system is necessary. Criteria may have to be set as to the speed of the compilation time, the run time, or both. Valid data on Ada compilers in this area is difficult to obtain as it is fairly easy for the vendors to skew the results of their tests with the mix of code they use in it. An unbiased source of information may be obtained from the Performance Issues Working Group (PIWG) of the Association for Computing Machinery (ACM). Their data are obtained from ACM volunteers running the test suites that the PIWG develops. However, the compiler being considered may not have been tested within the same machine configuration as that being considered. The PIWG also does not do an analysis on the results of the tests; they just publish the data and let the users draw their own conclusions.

If a requirement to use a validated compiler does not exist for the application under consideration (as may be the case in an engineering simulation) one could consider a non-validated compiler for development needs. Often non-validated compilers claim faster compilation and execution times than validated compilers. But it is important to examine carefully any compiler that has not been validated or is not planned to be validated. As stated earlier, the validation suites do not test all of the system dependent features needed for simulation applications but they do assure that all aspects of the language which are not system dependent are tested. This may become vitally important for future modification or porting considerations of the developed Ada code. The compilation/execution speed advantages that may be realized initially need to be weighed against the cost of future redesigns or recoding.

Once the compiler criteria has been established, one more list needs to be established: the compromise list. What trade-offs are acceptable in both the compiler and cross-compiler? They may be related to the criteria mentioned above or the associated software tools which are included with the compiler. An automatic recompilation system may outweigh, in relative value, the compilation speed of a compiler, especially for very large applications. Or it may not be possible to get the bit packing desired in the cross-compiler but it does offer a source target code debugging capability. Different vendors are focusing on different aspects of their compiler systems. It is worth the effort to investigate their track records also if a vendor is promising some features which are not required now but are absolutely necessary for use in the future. (Asking for customers' names from vendors for this purpose is an accepted practice). Care should be exercised in the compromise as it is emphasised again that the compiler and cross-compiler are the most important software tools in the simulation development system.

Software Engineering:

Software engineering is the term applied to the activity of creating software in a disciplined and consistent way. Ada's rich set of constructs and capabilities give the software engineers the ability to create a software solution which maps more accurately the problem domain it is addressing than other higher order languages which incorporate operating system calls or low-level assembler routines to effect the same solution. Simulation tackles complex systems and this fact coupled with the enhanced capabilities Ada offers requires a well thought out methodology for designing simulation software.

There are several software development methodologies available. Few, if any, cover all the aspects of designing software from requirements through testing. Most of the software design methods embodied in these methodologies are either top-down structured, data-structure, or object-oriented design. The object-oriented method seems to be emerging as a

favorite although most successful projects seem to be employing a combination of methods.

Another aspect to designing simulation software in Ada is what to do with all the developed FORTRAN code. Three options are possible: incorporate it, convert it, or re-design it in Ada. The first option is dependent on the Ada compiler one chooses. If it supports the pragma INTERFACE to the FORTRAN language, it is possible to use most of that code with minimal rewrites. The second option is the least desirable of the three. It implies purchasing a FORTRAN to Ada software conversion tool. There are a number of them in the commercial market. However, the Ada code which emerges would neither resemble well designed Ada modules nor understandable FORTRAN. The last option is the most desirable since the final product is a simulation code in one language. It may, however,also be the most time consuming and expensive and may not be feasible initially. Whatever option is chosen, incorporation of FORTRAN code needs to be taken into account in the design methodology.

SIMULATION CONFIGURATION

Figure 1 shows the McDonnell Douglas Helicopter Company multi-ship configuration arrived at after reviewing all the issues discussed earlier. As mentioned earlier, the approach was to use a distributed processing system for real-time simulation. The configuration uses existing processors and FORTRAN and other code as they existed. All computing expansion is achieved through microprocessors thus permitting a low cost and very affordable solution. This configuration also permits the main goal of developing all new software in Ada via a development system which is cross-compiled to the target system. An existing super-mini computer with compiler and crosscompiler provide the main software

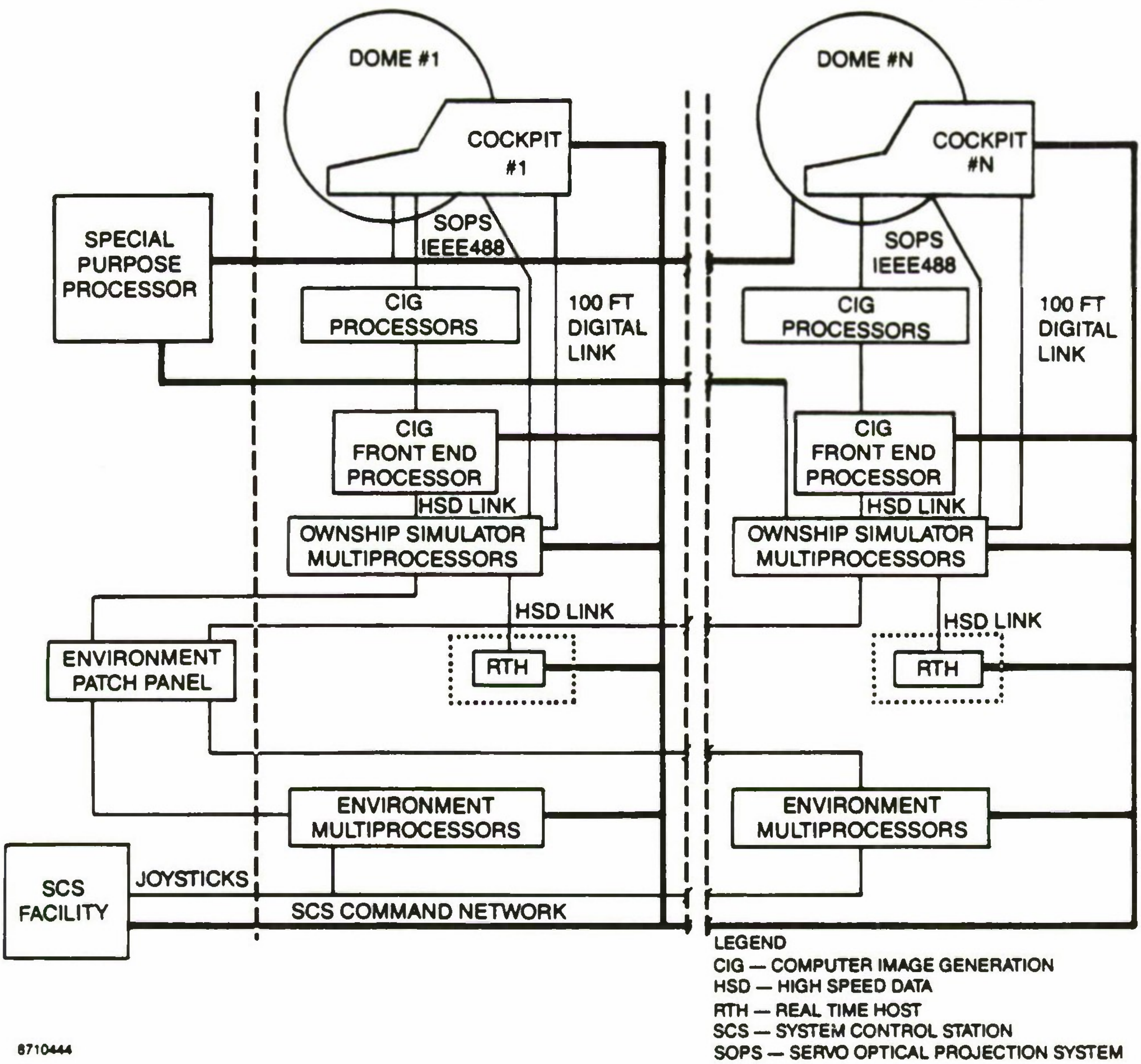


Fig. 1 Simulation Architecture

development capabilities. The configuration also permits gradual redesign of FORTRAN code to Ada and phasing in of processors to execute the redesigned code. PDLs, automatic document generator, and language sensitive editor are the primary software tools. These allow the development configuration to support high software productivity. Most importantly it forces a modular software/hardware approach in the long run.

LESSONS LEARNED

The McDonnell Douglas Helicopter Company simulation department has successfully moved into developing software in Ada, but this was not achieved without some difficulties. There were a number of lessons learned along the way.

Due to the fact that many Ada compiler and cross-compiler implementors are not quite "there" with all the features necessary for simulation software it is important to decide whether Ada will be the simulation language before investment is made into software and hardware for both the development and target systems. If money is invested in target systems before investigating the cross-compilers and debug tools, the advantages expected from a faster and more economical CPU may well be nullified by the fact that there are few, if any, good cross-compilers for the target system that has been purchased. As stated previously, development systems need enhanced capabilities when developing with Ada. Vendors who supply the software packages which allow this enhancement have not made these packages for every operating system. The development system and the software packages have to be considered as a whole and not in piece-meal. It is nearly impossible to retrofit software packages to any existing machine.

Designing software in Ada needs more training than that which is offered by the compiler vendors as part of the product they sell. Just knowing the syntax of the language is not enough. As stated earlier, a methodology for design does need to accompany the learning of Ada. Otherwise the price paid is creation of non-reuseable code if design issues are ignored in developing simulation Ada software. Unfortunately just selecting a methodology may not be enough. The methodology chosen may not fit the type of software that is being developed. One approach to this dilemma is to evaluate the PDL or code which is being produced by this methodology as soon as possible. If it is too complex, try another methodology or modify the existing one.

In the area of compilers and cross-compilers, a number of lessons were learned. On the issue of a validated versus a non-validated compiler, there is no question - use a validated compiler! This does not guarantee all the capabilities needed for simulation applications but it does mean the compiler has been extensively tested by validation suites and there will be fewer problems than with a non-validated compiler. The next lesson is that validation by itself is not enough. The buyer must be aware of what implementation dependent features of the LRM is needed for their applications. Knowing Chapter 13 and being able to understand Appendix F of a compiler vendor's manual is very helpful. The final lesson is to determine what compromises are acceptable, since it may not be possible to get all the features desired in a compiler. In the same vein, it is helpful to consider which vendor is making progress in the areas of capability we require, even if they do not have these capablities now.

CONCLUSION

This paper has reviewed the issues in implementing Ada in a multi-ship simulation organization. The issues were discussed in the light of MDHC's experience in achieving the transition to ADA. The advantages and disadvantages of Ada were examined. Technical issues such as hardware performance and requirements, software development environment, software tools, Ada compilers and cross-compilers, and software engineering were considered. MDHC's simulation configuration was discussed and lessons learned were presented. It was pointed out that the most important aspect is that all these issues must be examined in totality before any commitment is made to purchase hardware or software.

ABOUT THE AUTHORS

Dr. Ramachandran is a Senior Staff Engineer with the Engineering and Training Simulation Department. He has been involved in the setting up of the new Simulation Laboratory and the Apache and LHX simulator programs. His simulation experience covers a wide range of aircraft, helicopters and missiles.

His previous experience includes Goodyear Aerospace, Dynamics Research Corporation and NASA Langley Research Center. He has authored over twenty articles in the fields of simulation, aerodynamics and control systems. He received his B. Tech from the Indian Institute of Technology and M.S. and Ph.D. from the University of Cincinnati.

Mr. McCabe is presently a Systems Engineer with the Simulation Department and is involved in Ada Implementation for Simulation. He received his BSEE and MSEE from Arizona State University. He has many years of computer hardware and software experience.

INSTRUCTIONAL TECHNOLOGIES FOR EMBEDDED TRAINING

RICHARD E. REYNOLDS, Ph.D.
Human Factors Division
Naval Training Systems Center

KENT E. WILLIAMS, Ph.D.
Institute for Simulation and Training
University of Central Florida

ABSTRACT

It has been proposed that embedding training in operational military weapon systems can aid in achieving the goal of improved readiness. An analysis of embedded training goals and the potential contribution of embedded training toward enhancing personnel readiness was conducted. The emphasis in this specific project was the instructional technology requirements for embedded training, as opposed to the numerous engineering requirements relating to safety, reliability, etc. This analysis indicated that the advantages of shore-based training, particularly with respect to instructor functions, could be compromised in the embedded training environment. On the other hand, the fidelity and accessibility of training would be promoted by embedding training in operational equipment. To overcome this potential compromise of instructor functions, an evaluation of four instructional features, which could be implemented in the embedded training software, was undertaken. The instructional technologies under examination include: automated adaptive instruction, automated expository feedback, intelligent platforms, and simulation of missing team members. This paper will discuss the completed initial analysis and describe the research in progress. Initial data collected shall be summarized at the conference presentation.

I. INTRODUCTION

The military strategy supported by the United States is based upon a wide range of potential conflicts. This "Spectrum of Conflicts" ranges from a peace time show of military presence to strategic nuclear war. This translates into the requirement to counter enemy initiatives from air, land, and sea. With respect to the Navy, this counter capability becomes the application of Navy tasks such as antisubmarine warfare, antisurface warfare, antiair warfare, counter command and control, mine operations, amphibious operations, strike operations, sealift operations and special operations.

The primary threat today is represented by growing Soviet forces. To counter this Soviet threat, the United States has developed and implemented high technology to train personnel to effectively deploy their tactical systems. With the increasing complexity of these systems, the demands required of personnel to effectively apply U.S. tactics have increased. Readiness of personnel and systems is necessary to implement U.S. strategy. Of particular concern to this project is the training component of the personnel readiness factor.

Perishability refers to a degradation in the individual's ability to apply knowledge through behavior, either cognitive or psychomotor. For example, in October, 1985, COMNAVSURFLANT reported on an exercise relative to the Level of Professional Readiness (LPR) of Electronic Warfare (EW) specialists. The report indicated that the LPR did not improve from initial performance following "A" school through to retirement. This means that the skills learned in "C" schools and traditional OJT following "A" school perish over time and, in general, are not maintained or further developed as a function of job activities.

To attempt to counter this degradation in skill proficiency, the Navy requires additional training in those skill and knowledge areas where there is infrequent application of specific behaviors. Training systems of all types, ranging from shore-based, full mission simulators to desk top microprocessors which provide environments sufficient to train part tasks, are available to provide training. Such systems, however, are either not easily accessible or lack full mission capabilities. Another strategy has involved the development of pierside trainers. These are mobile containers which are trucked to the pier beside a docked ship and stimulate the equipment on board ship. Although the fidelity issue is nicely handled, problems with accessibility still remain, since it takes considerable time and numerous personnel to make all the necessary interfaces to the vessel's equipment, and a limited number of trainers must service an entire class of ships.

It has therefore been proposed that high fidelity of simulation and direct accessibility to trainers comes with a concept called Embedded Training (ET).

II. SHORTFALL

There are numerous definitions of embedded training but for now we shall refer to that provided in the DRAFT OPNAVINST on Embedded Training paragraph 4.1., Task Force on Embedded Training and the Flag-Level Steering Group on Embedded Training, 14 Nov. '85. "Embedded Training is training that is provided by capabilities built into or added onto operational systems, subsystems or equipment to enhance and maintain the skill proficiency of the fleet."

Much discussion, analysis, and research has been generated with regard to the engineering requirements for embedding training sub-systems in weapon systems under development and planned for the future. These engineering requirements emphasize safe reliable equipment with lock-out to weapon firing, immediate transfer from training mode to full system operational mode on demand, computer sizing, simulation vs. stimulation, strap on vs. full integration, etc. Other hardware technologies identified under engineering requirements for ET include such capabilities as voice recognition and computer generated imagery. The emphasis of this specific project is the instructional technology requirements for ET as opposed to the numerous engineering requirements.

III. APPROACH

Analysis of information published or available on 38 training systems identified by various sources as "embedded" (Table 1), as well as the wider literature on training technology, has led us to believe that most of the features of shore based training that are lost when training becomes embedded relate to the dimension of instructional technology. By embedding training in operational systems on a platform, many of the advantages of shore based training are removed. For example, one is now faced on board with the problems of

* providing a structured educational environment for training skills and knowledge
* providing interaction between students and instructor
* the absence of a large cadre of qualified instructors to assess performance, play other roles in scenarios, control targets, etc.
* the current absence of techniques and facilities for storing trainee relevant and unit relevant data for later analysis and administrative use.

The earlier mentioned definition of embedded training emphasizes equipment (i.e., built in, operational systems). One can then infer that instructional technology is a forgotten dimension. Indeed, our analysis of systems called embedded trainers, summarized in Table 1, shows that only a few trainers called "embedded trainers" have any instructional technology. Typically, the systems encountered are primarily signal generators or stimulators of some type, developed to present the exercises. This tendency is found to be true throughout the services. How will readiness be preserved in the absence of a major component of training, the instructor?

In reviewing the literature on instructor activities with respect to simulator based training, we found that a considerable portion of instructional activity is allocated to: creating, selecting, and modifying scenarios, monitoring trainee performance; simulating voice communications of missing personnel or teams; maneuvering platforms, modifying the sequence of exercises to provide scenarios which best develop trainee, subteam, or team weaknesses, and providing briefing and debriefing material. There is evidence to suggest that these kinds of activities may be provided automatically by incorporating expert systems technology and intelligent planning strategies to the course of instruction. Given the application of such technology to embedded training, the problem conditions which exist as a result of translating shore based training to embedded training can potentially be overcome. That is, those activities making up a major portion of instructor functions can be automated on board a platform.

Ageis Combat Training System (ACTS).
AN/TPQ-29 Improved Hawk Missle System (IHAWK).
AN/TSQ-73 Missle Minder Command and Control System(MMCCS).
Automatic Detection and Tracking Simulator (ADTSM).
Combat Control System MK-1 Training Mode.
Combat Team Operational Readiness Program (CTORP).
Carrier Air Control Center Shipboard Target Simulation System
Combat Simulation Test System (CSTS).
Electronic Counter Measures (ECM) Generator.
Guided Missile Simulator.
Guided Missile Training Round (GMTR).
In Flight Training (IFT) for F-14, AN/AWG-9.
Lesson Translator (L-TRAN) for NTDS.
LHD-1 Combat Simulation Test System (CSTS AN/SSQ-91).
On Board Simulation (OBS) for F-15.
On Board Electronic Warfare System (OBEWS).
On Board Trainer (OBT AN/SQS-T6).
Operational Readiness Assessment and Training System (ORATS).
Own Ship Motion Simulator (OSMOS).
OTS-V3R10 for AN/SLQ-32.
Performance Measuring Equipment (PME) for AN/SQQ-23.
Radar Recorder (RACOR).
Radar Video Recorder (RAVIR).
Radar Environmental Simulator System (RESS), AN/USQ-93.
Radar Proficiency Simulator (RPS).
Radar Video Simulator (RVS).
Radio Frequency Test Target Generator (RFTT).
Silverbox 2/WLR-1.
Submarine Operational Readiness Assessment and Training System.
Sonar Target Signal Simulator (STSS).
Simulated Target Training Program (STTP).
System Evaluator, Trainer SEAT.
Tactical Modular Display.
Tactical Proficiency Program (TPP).
Troop Proficiency Training (TPT).
Training Surface to Air Missle (TSAM).
Video Signals Simulator.
World Wide Military Command and Control System (WWMCCS)

Table 1. "Embedded Training" Systems Examined.

Fortunately, over the past ten years, technology has been implemented in various trainers which has automated some of the instructional and instructor support functions. These instructional technologies assist the instructor in conducting his activities during an exercise and, in some cases, perform an instructor's activities automatically. As several reviews have recently proposed, techniques for improving the modeling of instructor functions within the software of training systems are available employing techniques of cognitive science and artificial intelligence (Sullivan, Roth, Chinzoff & Bogner, 1986). It is, therefore, proposed that many of the problem conditions which emerge from embedding training, specifically as they relate to instructional activities, may be overcome by applying state-of-the-art software technologies.

Four technologies, specifically (1) adaptive computer-aided instruction, (2) automated expository feedback, (3) intelligent platforms, and (4) missing team member simulation, were chosen for further development and evaluation. Research and development was limited to these four based on our determination of criticality of need, resources available, maturity of hardware/software technology, and probability of success in developing technologies that can be quickly transitioned to a wide variety of operational systems under development.

Adaptive Computer Aided Instruction

Adaptive computer aided instruction allows the trainee to select the starting point for instruction and then assesses student strengths and weaknesses. This assessment is a continuous process. Once the system has determined initial strengths and weaknesses, it will automatically plan a course of instruction in order to present the student with exercises which best focus upon the students' specific strengths and weaknesses. During the course of this plan of instruction, the system also evaluates how rapidly the student is overcoming his weaknesses and whether or not specific strengths interact with new material in a deleterious way. The system, given this information through continuous assessment of student performance, can replan the course of instruction as needed. Consequently, the selection of exercises and the sequence or plan of exercises presented to the student is adapted to individual student strengths and weaknesses. These student plans can be saved and stored in memory so that later analysis can be conducted in order to identify any particular difficulties students are having when moving from less difficult to more difficult exercises. This information could then be used to modify the knowledge base of exercises at a later time. While numerous variations of this technology have been developed in intelligent tutor prototypes, it has not been tried, to date, in military ET systems.

Automated Expository Feedback

Automated expository feedback is a phrase which has been coined for this project. It applies primarily to rule utilization tasks as in decision making. The purpose of this form of feedback is to expose the error made on the part of the trainee by identifying specific preconditions which were not attended to or which were erroneously emphasized, consequently triggering inappropriate actions. This form of feedback employs expert systems technology and knowledge engineering techniques to simulate instructor actions by formulating a prescribed rule base for decisions and a search strategy to assess the application of specific decisions given a specific set of scenario conditions. The objective of this form of feedback is to promote the development of timely and accurate decision making within the constraints imposed by tactical doctrine.

Intelligent Platforms

Intelligent platforms serve to promote training by enhancing the realism with which targets maneuver in a scenario and by removing the requirement for an instructor to maneuver numerous targets within a scenario. These intelligent platforms are essentially expert system modules which test the states of the scenario and apply specific rules which are transferred into tactical actions. Incorporating these targets in exercises provides the trainee with greater realism in scenario conditions. Obviously the benefit of increased fidelity of trainer console interface is rendered worthless if the training scenarios do not provide comparable fidelity in terms of tactics displayed by targets in the training exercises. The skill level of targets can also be modified to vary the degree of difficulty of the scenario.

Missing Team Member Simulation

Missing team member simulation also involves the application of expert systems technology to training. This method removes the necessity for subteam members to actively participate in team training exercises, reducing the manpower requirement to conduct team training. More importantly, the application of this technology allows one to control the level of expertise of the simulated member(s). By doing so, specific criterion levels of performance can be maintained such that expectations on the part of the trainee participating in exercises can be directed toward high performance criteria. With each member of a team being trained in an environment with a common reference of expectations of other team member performance, team member performance in actual combat would be improved since all members would expect high criterion levels of performance. This is based upon empirical evidence which has demonstrated that team performance was highly dependent upon individual member expectations of other team member capabilities for communication and coordination tasks (Crowe, Hicklin, Kelly, Obermayer, & Sutzer, 1982). These tasks are the predominant activities of teams and subteams.

IV. STATUS AND PLANS

Having completed our review and analysis of the literature, and chosen the above-described technologies for further evaluations, we are currently (at the time of this writing) completing implementation of several intelligent platforms and a missing team simulation for preliminary evaluation. The evaluations are to be conducted on the command and control research testbed, currently housed in the Human Factors Division at the Naval Training Systems Center (NTSC). Evaluations of these technologies will begin during the summer of 1987, and results should be available for discussion at the time this paper is presented, in November 1987. Following evaluation of these capabilities, those technologies judged successful will be considered for modification, limited implementation and evaluation on the embedded training element of the

NAVAL TACTICAL DATA SYSTEM (NTDS), known as the Lesson Translator (L-TRAN). L-TRAN programs are currently supplied to over 150 major surface combatants and several shore-based schools from the L-TRAN Project Office at Fleet Combat Training Center, Pacific.

The adaptive computer-aided instructional module referred to above has also been designed and is under development by the University of Central Florida Institute for Simulation and Training (under contract with ONR and NTSC). A product of this task will be lesson design specifications to guide implementation of adaptive lessons for the L-TRAN. Based on these guidelines, experimental adaptive lessons will be written, in conjunction with the L-TRAN project office. These lessons will be debugged and a preliminary evaluation completed on an L-TRAN emulator housed at the Human Factors Division of NTSC, using Naval personnel from the various Service School Command schools in Orlando as subjects. Assuming a generally successful outcome from this evaluation, the adaptive strategies will be revised, implemented, and evaluated in a limited number of operational settings.

In summary, the instructional technologies defined above serve to automate many of the activities of an instructor and support personnel as well as provide additional realism to exercises. Given the emphasis upon fidelity as witnessed by the trend toward embedded training, it seems only consistent to provide more realism in terms of target maneuvers. Anticipated target maneuvers and tactics make up a major portion of what must be learned in tactical decision-making training. These technologies, although capable of being implemented, have not strictly been evaluated to determine their impact upon acquisition and retention of skills and knowledge. Since the instructional dimension is a significant component in the training equation, it is important to evaluate the impact of these technologies on what is learned, how rapidly learning takes place, and how well learned skills and knowledge are retained. Furthermore, by controlling the consistency of expertise of instruction by employing concepts such as expository feedback, transfer of expert information can be provided without the interference and inefficiency of learning by trial and error. This technology filters out bad instances in tactical decision-making and guides trainees along successful solution paths. The objective of this project effort is to both develop and evaluate the candidate instructional technologies described above.

REFERENCES

1. Crowe, W., Hicklin, M., Kelly, M., Obermayer, R., & Sutzer, W., Team Training Through Communication Control (NAVTRAEQUIPCEN, Report Number 80-C-0095-1), Vreuls Research Corporation, Thousand Oaks, CA, 1982.

2. Malehorn, M. K., Evaluating the Requirement for Exploratory Development on Embedded Training (Contract: N61339-82-D-004 D.O. 0023), Eagle Technology, Inc., Arlington, VA, 1986.

3. Polzella, D. J., Aircrew Training Devices Utility and Utilization of Advanced Instructional Features: Phase II - Air Training Command, Military Airlift Command, and Strategic Air Command (Contract: F33615-81-C-0005), University of Dayton Research Institute, Dayton, OH, 1983.

4. Task Force on Embedded Training and the Flag-Level Steering Group on Embedded Training, 14 November 1985.

5. Sullivan, G. K., Roth, J. T., Chenzoff, A. P., & Bogner, M. S., Tri-Services Review of Existing Embedded Training (ET) Systems (Contract: MDA-903-85-C-0078), U.S. Army Research Institute, 1986.

BIOGRAPHICAL SKETCH

Dr. Richard E. Reynolds is the technical team leader for Embedded Training and Command and Control Training in the Human Factors Division of the Naval Training Systems Center. As such he is responsible for coordinating all related projects within the division. He holds a Master's Degree and Ph.D. in Experimental Psychology from Miami University. He was formerly an associate professor of Psychology at Edinboro University of Pennsylvania. Dr. Reynolds has over 25 publications and presentations in the areas of Experimental Psychology and Human Factors.

Dr. Kent E. Williams is research manager for the University of Central Florida's Institute for Simulation and Training. Dr. Williams has been involved in applications of Artificial Intelligence and Cognitive Psychology for the enhancement of training systems. Dr. Williams holds a Master's Degree in Neurobehavioral Psychology from Connecticut College and holds a Ph.D. in Human Information Processing from the University of Connecticut. Dr. Williams has over 50 publications in the areas of Artificial Intelligence, Training and Simulation and Human Factors.

IDENTIFICATION OF CRITICAL INSTRUCTIONAL SUPPORT FEATURES FOR EMBEDDED TRAINING IN THE SHIPBOARD ENVIRONMENT

William F. Jorgensen
Eagle Technology, Inc.
Winter Park, FL

Daniel J. Dwyer
Naval Training Systems Center
Orlando, FL

Michael Oberlin, Ph.D
Naval Training Systems Center
Orlando, FL

Brenda J. Hoskin
Eagle Technology, Inc.
Winter Park, FL

ABSTRACT

Embedded training has long been considered a potentially efficient training concept which could provide meaningful use of available time and resources to maintain skill proficiency levels or teach new skills while on the job. A major problem is the development of embedded training which can be used effectively by a single user and which will provide management and control of the training environment. Factors such as varying levels of training complexity and measurement of trainee performance are important training issues, and must be included in the training design. The Human Factors Division at Naval Training Systems Center is presently engaged in embedded training research using the AN/SPA-25G radar repeater as a test bed. The newly developed AN/SPA-25G radar display is a computer controlled console which can be used to automatically compute calculations such as intercept courses and speeds, closest points of approach and many other similar functions formerly requiring the use of maneuvering board procedures. This embedded training project is using the capabilities of the AN/SPA-25G radar repeater and innovative scenario generation software to develop both a training process and the necessary instructional support features which will deliver and manage the radar operator training onboard ship during routine operating hours. Training programs currently being developed include equipment proficiency training for newly assigned operators and for more experienced operators, task component training (practice of specific skills within a given task) either on the PC itself or on the AN/SPA-25G radar display, and scenario training with multiple targets.

The driving force behind the successful implementation of embedded training lies in the reduction of instructor workload while still providing quality training. One way that this can be accomplished is through a judicious application of key instructional support features. This paper discusses the methodology used in identifying 11 critical instructional support features necessary for successful embedded training in the AN/SPA-25G radar repeater and defines each of these important features.

INTRODUCTION

During deployment, Naval personnel are expected to maintain skill proficiency levels through on-the-job training (OJT). While OJT has been demonstrated to be a successful method for developing job-related skills in certain settings, the job environment is often not a good learning environment (Goldstein, 1986). In the Navy, the opportunity for OJT is often precluded due to shipboard constraints such as (1) high workload, and hence limited availability of qualified personnel who can fill the role of "onboard instructor;" (2) matching student availability with instructor, equipment, and "live" aircraft availability; and (3) operational commitments of equipment and personnel for mission requirements vice training needs. As a result of these limiting factors, maintaining certain high skill levels in a shipboard environment presents a formidable challenge.

The concept of embedded training (ET), has long been considered a promising solution to shipboard training problems. ET may be conceived of as a training capacity which is designed into an operational system. It has been defined by the Navy as "training that is provided by capabilities built into or added onto operational systems, subsystems, or equipment to enhance and maintain the skill proficiency of fleet personnel" (Department of the Navy, 1985). ET promises to reduce onboard instructor workload requirements, if training scenarios utilize sound instructional support features, and the training is integrally designed into the system. However, the features which make ET easy to use and a valuable training tool have rarely been designed into the system.

The Naval Training Systems Center (NAVTRASYSCEN) is currently engaged in a research program aimed at developing, implementing, and evaluating an ET capability within the Radar Display and Distribution System (RADDS). RADDS, which is being developed by the Naval Sea Systems Command (NAVSEASYSCOM), is composed of three major components: (1) the AN/SPA-25G radar repeater, (2) the SB-4229 switchboard, and (3) the CB 3989 converter. These three components incorporate state of the art radar display technology. The system is scheduled to be phased into the Navy over the next decade. The key component of this system is the AN/SPA-25G radar repeater, a solid state (except CRT) raster scan display that presents selected radar video from one of several basic radar systems. The AN/SPA-25G has six operational modes: Air Intercept Control (AIC), Navigation, Anti-Submarine Warfare (ASW), Anti-Surface Warfare (ASUW), Amphibious Assault, and Electronic Warfare (EW). The system is dramatically different from other non-Navy Tactical Data System (NTDS) radar repeaters in that the system software automates many of the computations traditionally performed by the operator (e.g., Closest Point of Approach, air intercept computations).

The ET research program is composed of several systematic steps which are intended to culminate in a self-contained training capacity within the RADDS. The intended training "packages" within this effort include equipment

proficiency training, component skill training, and mission scenario training. The major components of this research effort include: (1) performing a training needs assessment in order to determine precisely what type of ET can make the greatest contribution; (2) identification of the critical instructional support features necessary to support ET; (3) development of scenario control software which will provide the means for creating training scenarios; (4) development of the real-time/play software which will execute the training software, track targets, and maintain performance records; (5) development and implementation of the equipment proficiency and component skill training; (6) development and implementation of training scenarios; and (7) evaluation of the ET capability. This paper focuses on one phase of this research effort: the identification of the critical instructional support features necessary to support ET within RADDS.

THE PROBLEM

As mentioned previously, shipboard constraints often inhibit the opportunity for hands-on practice, a critical component of OJT. Without this opportunity, the skill levels of operational personnel may rapidly decay. This is particularly troublesome when one considers that students may spend in excess of four months in formal school training only to discover limited opportunities to practice these newly acquired skills upon reporting aboard ship. This section will address some of these problems which interfere with the opportunity for hands-on training aboard ship.

Availability and Workload of Onboard "Instructors"

Deployment of a ship typically centers on a set of mission requirements. These requirements have top priority. As a result, training is often relegated to a lower level of priority. This is especially true of training for inexperienced personnel. Senior enlisted personnel are often a driving force in achieving mission goals and most of their duties are focused on these goals. Consequently, they have limited time to devote to the role of "onboard instructor". Without guided supervision from an experienced individual, inexperienced personnel may find themselves caught between the need for hands-on training and the lack of supervision for training. Even if available, the supervisor who is an expert on the system may not be qualified in instructional techniques or strategies necessary for proper training.

Student Availability

Even if the problem of onboard instructor availability could be resolved, a problem concerning student availability may emerge. Very often, new graduates reporting to their first duty station are assigned collateral duties (e.g., mess cooking) for up to six months. During this period, little effort may be devoted to practicing their previously acquired skills, and as a result, these skills may degrade rapidly (McDonald, 1984). The issue is compounded when one considers the difficulty in matching student availability with instructor availability and, can become even more severe by adding additional constraints such as availability of equipment and "live" targets to use for training.

Operational Tempo

A third problem area concerns the changing operational tempo associated with deployment. There are usually prolonged periods during which at-sea operations are fast-paced, hectic, and highly stressful--factors which are not conducive to structured learning activities featuring learning principles such as self-pacing, feedback and remediation. Furthermore, these operational factors may contribute an element of danger in a learning environment; for example, when the inexperienced radar operator provides an incorrect bearing and range to a target posing a navigational danger.

Equipment Design

Finally, operational equipment is designed to meet operational requirements. If training needs are considered at all in the design process, that consideration takes a secondary role. Consequently, when the equipment is to be used in a training function, limitations may arise. For example, in the case of a radar repeater, it is obvious that the primary goal is to display live aircraft, surface vessels, and landmass. However, when live aircraft and other targets are not present, nothing is displayed and very limited training (if any) can occur.

The use of equipment stimulation as a training tool has been used on some operational systems. For example, radar stimulators, which are designed to support system alignment and calibration (by displaying and checking synthetic targets with known ranges and bearings), have been adapted for use in training. Typically, these configurations fail to achieve their full training potential due to the lack of instructional features needed to support the training environment.

A potential solution for reducing problems encountered with shipboard training and lessening the reliance on nonstructured on-the-job training focuses on ET. By coupling radar stimulation technology with critical instructional support features, ET may reduce many of these shipboard constraints. For example, self contained training scenarios coupled with instructional support features such as augmented feedback, scenario initiation and control, and performance monitoring may serve to combat the problem associated with onboard instructor availability. Features such as replay/playback, recordkeeping, and performance measurement may contribute to reduced instructor workload. Additionally, the capability to display synthetic targets on the radar display via a target generator will lessen the reliance on "live" targets and will ensure that training is available at the students' convenience (e.g., after hours, slack time, etc.). The training can therefore occur without being impacted by the operational tempo and at a time when both the student and the equipment are available.

APPROACH TO PROBLEM RESOLUTION

In the shipboard environment, the typical scenario surrounding training activities includes the following: there are technicians who require training, supervisors who can, to some extent, guide the training activity, and limited availability of operational systems upon which to perform the training. The major problem is the coordination of these requirements and availabilities to accomplish meaningful training.

To fully understand how embedded training may be an effective solution to the problems of shipboard training, it is necessary to further examine the basic premise for a key element of ET - instructional support features. Instructional support features may be defined as characteristics of the equipment or trainer (hardware) which can be designed or programmed to control and/or present many elements of the training activity. They are important because they represent means to address two areas which may hinder effective training. These are instructor availability and the training technique or strategy employed. The significance of instructional support features in providing "programmed", structured and consistent training strategies is that their use can ensure control over the four essential components or steps inherent in the learning process. These learning components include the stimulus or cue (element to be solved or learned); the response (student reaction to the stimulus); the feedback (meaningful information concerning the appropriateness of the response); and the selection of the next activity (what the student should do next). These four aspects of the learning environment must be controlled by someone or something. All four ingredients must be present in the learning environment, regardless of who has control over them. Instructional support features are not capabilities that make learning easier, rather they are processes that are controlled by the equipment or trainer, which could be controlled by the instructor. Thus, instructional support features in the embedded training environment serve to reduce required instructor/supervisor time during training, facilitate the delivery of sound instructional strategies and provide the ability to measure the performance and track progress of the student.

Selection of the correct and necessary instructional support features requires a complete understanding of the operational environment into which the training procedures will be placed. Therefore, it was necessary to examine the activities of the operations specialist (OS) in the combat information center, how they utilized the AN/SPA-25G, what training they had received prior to shipboard duty, and that training necessary to remain proficient with the equipment. After this examination, the time requirements of the supervisors, along with the restriction and availability of trainee time and equipment had to be determined. Because instructional support features are designed to utilize the capabilities of the system that they support and to address the specific training requirements of the target environment, it is necessary to first examine and identify these elements to form the basis for the selection of the features and the development of the total embedded training package. Thus, the approach taken was to first determine the kind and type of embedded training to be done, identify those features of the environment which had to be augmented or controlled for successful training, specify the instructional feature characteristics of the training system necessary to support embedded training, and conduct a trade-off analysis which would help select the instructional support features and determine the complexity to which they should be developed. This analysis process was comprised of three initial steps: identify the basic type or area of training to be addressed, develop a list of potential instructional support features, and define the specific training requirements for which the features would be used. These three issues will be addressed in turn in the following paragraphs.

Type Of Training

The first step in the ET analysis was to identify the primary areas which training should address. This process began with the examination of three basic operator task areas. These were air intercept control (AIC), anti-submarine air control (ASAC) and navigation/piloting (NAV). The study of these basic tasks with subject matter experts (SMEs) supported the point-of-view that, contained within these tasks were training requirements in the areas of equipment, task and mission performance.

Further discussions with fleet SMEs and technical school instructors elaborated on these training requirements and resulted in three distinct training areas which were defined as follows:

- o Equipment proficiency training - the maximum and correct use of the system for the task at hand. This was further defined by the SMEs as correct use of all operational capabilities of the system to achieve maximum usefulness in performance of the job. An example is the use of the Closest Point of Approach (CPA) function in track mode of the AN/SPA-25G radar repeater.

- o Task component training - the mastery of one or more elements of a task. This means that a person should be able to practice a single critical part (or parts) of a task in isolation from the remainder of the task. An example of task component training is the practice of controlling a maneuvering aircraft during an ASAC problem task.

- o Mission Scenario training - refers to the use of one or more complete tasks to create a scenario which reflects the activity of a person (or persons) in an actual operational environment. For example, controlling an interceptor from Combat Air Patrol (CAP) station to intercept and back to home base.

Each of these types of training should be designed to accommodate low to high entry level personnel.

Instructional Features

Once the above general areas were identified as training requirements, a list of potential instructional support features required to insure success in the embedded training environment was developed. Examination of reference documentation (Hritz, Harris, Smith and Purifoy, 1980) as well as historical data identified a large population of potential instructional support features to be considered for implementation. Generally defined, these features can be described in terms of four categorical types: monitoring instructional features, feedback instructional features, stimulus instructional features and miscellaneous features. This pool identified all possible features from which those most feasible would later be selected.

Training Requirements

The third step of the analysis entailed a critical examination of the specific training requirements for operational personnel. The methodology employed to define the training was a structured interview process during which the SMEs were questioned at length on critical factors, such as:

- o Training requirements aboard ship - difficult training activities to accomplish, most training intense tasks, proficiency training needs, significant training needs, most often performed tasks.
- o Radar display requirements - what are critical radar characteristics to replicate in training, what needs to be most frequently replicated from the actual environment.
- o Fidelity issues - what are distinctive targets, accuracy of displays, reality aspects of environment.
- o Criticality issues - what is essential to job performance, impact of tasks if poorly done, what training is not done due to safety considerations.
- o Environmental issues - availability of supervisory help, performance measurement considerations, availability of training time, equipment availability.

These and other related questions were the catalysts to data collection interviews with shipboard, staff and technical school operations specialists. The results of these discussions led to the conclusions that for embedded training to be successful, supervisor's time to manage training must be minimized, training performance must be measured and improvement recorded over time, and the features required should have minimal impact on equipment. In addition, the issue of cost was deemed a critical consideration, particularly in the design specifications describing the complexity of the features, e.g.: use of sound powered phone circuits for practicing communication procedures versus the use of interactive voice recognition and digital voice generation to simulate a communications circuit.

The preceding three step analysis process and the resultant issues identified provided the basis for an initial selection of the most appropriate instructional support features. Based on the analysis, seventeen (17) features were chosen as those which were most acceptable and which should be further examined to provide for support of ET. These features are as follows:

- o Scenario Control - this feature is to be software containing actual preprogrammed scenarios which are to be linked to the target generator which, in turn, sends appropriate signals to the AN/SPA-25G replicating the actual environment.
- o Student/Instructor Cueing - visual cues consist of various messages which are either printed on the screen in graphics for help or information or are presented in the form of lights or buttons flashed on and off. Audio cues are directions such as those directing one to track or initiate intercepts and can be verbal when the scenario is initiated.
- o Target Control - In the operational environment, targets or ownship must respond to the operators' direction or recommendation. Thus, in ET, the target will have to be programmed to follow the trainee recommendations for Course/Speed changes.
- o Signal-To-Noise Ratios - In the operational environment, the operator has to contend with static or noise in both visual and audio cues. This feature can replicate "noise" which can be controllable by programmed scenario difficulty.
- o Record Keeping - this feature stores the trainee's performance scores (by name, date, and SSN) and maintains a history of accomplishment for each trainee.
- o Voice Recognition - for this feature, the computer is programmed to recognize a student's voice and accept it as the only voice to respond to. When these commands are accepted, the computer directs the designated target to do as instructed by the student.
- o Voice Synthesis - in this feature, the computer generates a voice to represent the responding (of a target) to the verbal commands of the student.
- o Performance Measurement - this feature senses all of students actions, compares them to a standard and develops a performance profile during the exercise.
- o Verbal Recording - this feature stores all voice communication within each scenario.
- o System Monitor - this feature (coupled with the ability to sense all student actions) evaluates those actions and provides feedback regarding the appropriateness of the action.

- o Sign In - this feature records the time the student initiated the training activity and records all data regarding performance in that students name.

- o Freeze Action - this feature alerts either the trainee or the instructor when the trainee makes a response that is so far off proficiency as to require special help.

- o Replay/Playback - this feature is the ability to replay entire scenarios including the student's actions previously recorded.

- o Fast/Slow Time - this feature enables the speeding up or slowing down of a problem or scenario so as to limit non-productive wait time. This feature can also be used to reduce the time it takes to replay an exercise.

- o Reporting Devices - this feature is a device which can print a copy or present on a video screen the recorded student actions or scores.

- o Feedback - this feature provides performance information (knowledge of results) to the student after the completion of a scenario. This feature provides the student with information concerning the correctness of the student's responses. This information is then used by the instructor as a key element in the presentation of augmented feedback: information concerning how or why responses were incorrect and how to improve performance.

- o Rate Control Adjustment - this feature controls the rate and quantity of scenario cues which are presented to students. This feature can alter complexity and/or difficulty by increasing or decreasing the number or presentation rate of scenario elements such as landmass obstacles, targets, and environmental conditions.

<u>Trade Off Analysis</u>

Once the instructional support features were identified, it was necessary to evaluate each one based on fixed criteria and compare them to the various types of training required: equipment proficiency, task component and mission scenario. Features were evaluated by the following criteria for each training type:

- o Cost of feature - this was described in relative cost terms when compared at various levels of complexity for each feature.

- o Fidelity - degree of reality when compared to the actual system.

- o User acceptance - an estimate by SMEs of how well the feature or change would be accepted in the operational environment.

- o Training effectiveness - an estimate of the positive impact of the feature on the training environment.

The above evaluation of the features was further refined according to their absolute requirement in the training environment. They were placed in one of four categories:

- o Required for embedded training - without which training cannot occur.

- o Will minimize supervision - if used, the supervisor's time will be reduced, but training can occur without.

- o Desired but not necessary - does not have a significant effect on either supervision or training, but would be an assist to the trainer.

- o Nice to have - not needed, but would provide some attractive features.

The result of these exercises was a list of features which were rated highest in comparison to criteria and which were either required for training or minimized supervision. These eleven instructional support features were: Target Control, Freeze Action, Replay/Playback, Scenario Control, Feedback, Record keeping, Performance Measurement, Sign-In, Student/Instructor Cueing, System Monitor, and Signal-To-Noise Ratios. These eleven features are judged most likely to ensure success of embedded training on the AN/SPA-25G radar repeater.

CONCLUSIONS

Successful embedded training requires the judicious use of instructional support features to ensure the ease of use and the adequate training control necessary in the operational environment. This design is required due to limited training time, equipment and supervision. Embedded training for the AN/SPA-25G radar repeater must include software/hardware for implementation of these important system features.

Each of these critical instructional features will be considered in the next phase of this research effort - the development and implementation of ET scenarios. While these critical instructional features were determined via a systematic approach, and tapped the knowledge of SMEs, little or no empirical data concerning their contribution to the training function exists. Only through carefully controlled evaluations can the validity of these features be judged.

REFERENCES

Department of the Navy. Memorandum by the Chief of Naval Education and Training, Report on Embedded Training for Flag-Level Steering Group, Pensacola, FL, (November, 1985).

Goldstein, I.L. <u>Training in Organizations: Needs Assessment, Development, and Evaluation</u>. Belmont, CA: Brooks Cole, (1986).

McDonald, L.B. <u>Passive Acoustic Analysis Training Problems</u>, NAVTRAEQUIPCEN Technical

Report 83-M-0844-2, Orlando, FL: Naval Training Systems Center, (1984).

Hritz, R.J., Harris, H.J., Smith, J.A., and Purifoy, G. R. Maintenance Training Simulator Design and Acquisition - Handbook of ISD Procedures for Design and Documentation. Valencia, Pennsylvania: Applied Sciences Associates, (F33615-78-C-0019), (1980).

ABOUT THE AUTHORS

Mr. Jorgensen is the Senior Scientist for the Orlando office of Eagle Technology, Incorporated, and is responsible for the technical approach on Instructional Development projects. He is also manager of several training analysis projects, one of which is the AN/SPA-25G embedded training analysis effort. Mr. Jorgensen has been with E-Tech over seven years, during which he has managed analysis efforts for all four services. He is also responsible for the development, maintenance and evolution of E-Tech's automated ISD process called ETRAN, which has been used for several analysis projects, to include the current AN/SPA-25G effort. Mr. Jorgensen holds degrees in Instructional Design and Technology, Mathematics, Science Education and Media Design. He has authored papers for several conferences and reports in the field of vocational education and training.

Daniel J. Dwyer is a Research Psychologist in the Human Factors Division of the Naval Training Systems Center. He has over eight years experience in military training R&D in the areas of human factors front-end analysis, development of design guidelines for maintenance simulators, evaluation of visual display parameters for training and aiding applications, computer-assisted instruction, and embedded training. Mr. Dwyer received his M.S. degree in Industrial/Organizational Psychology from the University of Central Florida in 1979. Currently, he is working on his Ph.D. in Industrial/Organizational Psychology at the University of South Florida.

Michael Oberlin is a Human Engineering Psychologist at the Naval Training Systems Center in Orlando, FL. In addition to several efforts in the area of embedded training, Dr. Oberlin's primary work experience has been in the area of user-computer interface design. Projects have included developing user-interface software requirements for a team anti-submarine warfare trainer, designing an authoring system for generating radar operator training scenarios, and evaluating display parameters and user interface software characteristics on the AN/SPA-25G radar display system. Dr. Oberlin received his Ph.D. in Applied Behavior Analysis for Western Michigan University in 1982.

Brenda Hoskin is a Training Analyst at Eagle Technology, Inc. in Winter Park, FL. Ms. Hoskin has experience in the areas of embedded training, front-end analysis, individual training standard systems design, training performance research and CBI research and development. Ms. Hoskin received her M.S. degree in Industrial/Organizational Psychology from the University of Central Florida in 1986.

TEAMWORK FROM TEAM TRAINING: AN ASSESSMENT OF INSTRUCTIONAL PROCESSES IN NAVY TEAM TRAINING SYSTEMS

Ben B. Morgan, Jr.
University of Central Florida
Orlando, FL

Eduardo Salas
Human Factors Division
Naval Training Systems Center
Orlando, FL

and

Albert S. Glickman
Old Dominion University
Norfolk, VA

ABSTRACT

This paper presents findings from a cooperative research effort between the Center for Applied Psychological Studies of Old Dominion University, Norfolk, VA, and the Naval Training Systems Center, Orlando, FL. These studies of Team Evolution And Maturation (TEAM) are designed to investigate the development of teamwork during the training of operational Navy teams. Initial results are summarized in terms of a general model of the phases of team evolution and maturation, a "developmental" research perspective based on this model, prototype procedures for measuring team development during training, and data which provide empirical support for the model and measurement procedures. In addition, findings are presented which help to explicate the instructional strategies and processes employed in team training. The implications of these studies are discussed and recommendations are given concerning interventions for improving team training instructional technology.

INTRODUCTION

In a presentation to the 1985 Interservice/ Industry Training Systems Conference, Salas reported that the Human Factors Division of the Naval Training Systems Center had initiated a systematic R&D effort to address several problems associated with previous team training and team performance research (18). He discussed two new programs aimed at establishing guidelines for the training of operational military teams. One of those programs of research is being conducted by the Center for Applied Psychological Studies of Old Dominion University. This research has been designed specifically to investigate the processes involved in the development and evolution of Navy teams in training. The ultimate goal of these studies of Team Evolution And Maturation (TEAM) is to enhance the design of team training systems by (a) providing a greater understanding of the factors that influence the development of teamwork during operational Navy training, and (b) developing interventions to improve the instructional technologies used in Navy team training systems.

The focus of this research on the development of teamwork during team training is rather unique. Although previous authors have acknowledged the dynamic and "organismic" nature of teams (cf. 6, 8, 9, 13, 15), very little research has focused on the developmental processes involved in the time-dependent acquisition of teamwork skills. Most studies have involved fully-mature teams that have already developed the skills required in interacting and coordinating team performance activities.

The current research is based on the belief that a fuller understanding of the developmental patterns of the behaviors associated with effective intrateam coordination, cooperation, communication, etc. will contribute substantially to the enhancement of team training systems. The purposes of the current paper are to (1) describe this unique research program, (2) summarize results which illustrate the validity of the approach and its measurement procedures, and (3) present conclusions and recommendations that have relevance for the design of future team training systems.

THE "TEAM" METHODOLOGY

The Perspective

The developmental focus of this research is based on the assumption that effective team training will produce measurable changes in team behaviors that enhance the efficiency and effectiveness of teamwork. Thus, it is expected that teamwork will develop through several phases that begin with a loosely organized group of individuals and ends with a highly effective team whose members interact, coordinate, communicate, etc. in ways that are optimum for the performance of their assigned task(s). This perspective, which has provided the general orientation for the current research, is represented in Figure 1.

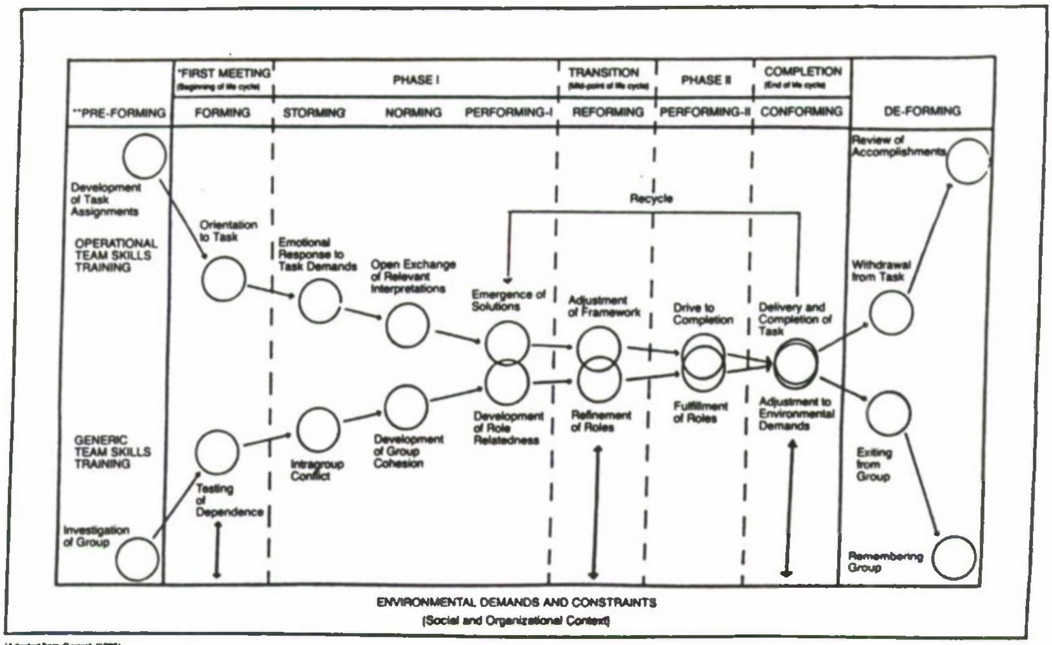


Figure 1. A Generalized Model of Team Evolution and Maturation.

Based on the suggestions of several previous authors (e.g., 1, 2, 3, 4, 5, 10, 11, 19, 20), this model of team evolution and maturation (i.e., the TEAM model) indicates that task-oriented teams evolve through a series of developmental phases (and, presumably, effective team training exercises will enhance the progression of teams through this evolution). Of course, different teams will begin at different stages of development and spend different amounts of time in various phases, depending upon the characteristics of the team members, their past history and experience, the nature of their task, their environmental context, the efficacy of their training, and other variables. In fact, although all the phases in Figure 1 have been identified by previous authors, it should not be expected that all teams will progress through all of these phases during the course of any specific training program. Nevertheless, the model suggests that it should be possible to document the development of teamwork from levels that are characterized by ineptness and exploratory interactions to the final levels of efficient and effective performance.

The model tracks two distinguishable kinds of team activities across the stages of evolution and maturation. As suggested by Tuckman (19), the first set of these activities (represented by the upper row of circles) is related to the development of skills involved in performing the team's assigned technical task(s). That is, a substantial portion of a team's effort will be devoted to the development of "operational skills" (see 7), such as those involved in understanding the task requirements, discovering the rules of performance, learning prescribed communication requirements, acquiring necessary task information, etc. On the other hand, teams also devote considerable effort to the development of "generic skills" (7; represented by the lower row of circles in Figure 1) that are involved in the development of team interactions, relationships, affects, and coordination. These activities include the establishment of roles, the development of cohesion, the development of team structure, etc. They are essential parameters in the development of successful teams. The TEAM model and its theoretical foundations are discussed in greater detail by Morgan, Glickman, Woodard, Blaiwes, and Salas (17).

The Procedures

In order to test the research perspective outlined above, a battery of data collection devices were developed to (a) measure team demographics (particularly the ranks--or rates--of the team members, and their levels of experience in the Navy and in their current assignment), (b) sample the development of behaviors that are critical to the maturation of teamwork, (c) assess changes that take place in the perceptions of team members concerning the team's knowledge and abilities, motivation, communication skills, coordination, etc., (d) estimate the levels of performance of the team members and the team as a whole, and (e) determine the instructional stategies and techniques used by instructors during team training. In the study reported here (the first of three such studies planned for this project), these instruments were used to measure the development of teamwork in 13 Combat Information Center (CIC) teams undergoing training at the Naval Gunfire Support (NGFS) simulator at the Naval Amphibious School, Norfolk, Virginia.

This research focused on CIC teams for several reasons: (a) the typical CIC team consists of eight team members, (b) CIC is the most critical subsystem of the NGFS activity, and (c) CIC performances require a substantial amount of intrateam interdependency, communication, and interaction. In addition, findings obtained with these teams offer high potential for generalization to many other Navy teams whose operations are similar to that of the NGFS CIC teams. Training of the CIC teams consisted of a one-half day orientation session followed by 3 1/2 to 4 1/2 days of simulation exercises. The simulator training is presented in five phases (Basics, Pre-midterm, Midterm, Post-midterm, and Final), and data concerning the development of teamwork were collected for each phase (at the end of each morning and afternoon training session). On-line performance (criterion) data were also provided by the School from scores on the Midterm and Final test exercises performed by each team. Team membership and position assignments remained the same throughout training.

Only a part of the results are presented here. These were generated primarily from data from a Trainee Self-Report Questionnaire (TSRQ) and semi-structured interviews with NGFS instructors; other results from this study are discussed by Glickman et al (12). The TSRQ is a modification of a similar questionnaire used by James, Gustafson, and Sells (14). Using a five-point Likert scale, each trainee completed this 21-item questionnaire at the end of each morning and afternoon session of training. The items measured the trainees' perceptions of the job knowledge, motivation, role clairity, experience, and training of the other team members, and the overall team's communication, cooperation, coordination, experience, training, and power relationships.

In the period during which the TSRQ (and other TEAM) data were being collected, NGFS training procedures were observed directly, and instructors were interviewed concerning the instructional processes that are employed in NGFS training. One purpose of this effort was to determine how instructors assess the training needs and performance capabilities of teams and how they select appropriate training strategies for use with a given team. The interviews were also aimed at identifying the decisions made by instructors, the instructional strategies and tactics that they employ, and the content and timing of feedback that they provide to trainees.

TEAM DEVELOPMENT

In order to examine the changes that take place in teams as they undergo training, data from the TSRQ were submitted to a series of exploratory factor analyses. Specifically, phase-to-phase transitioning and change were examined by combining TSRQ data from adjacent training phases and factor analyzing each of the resulting groupings of the data. Thus, each of the following four data combinations were factor analyzed: (1) Basics and Pre-midterm, (2) Pre-midterm and Midterm, (3) Midterm and Post-midterm, and (4) Post-midterm and Final. The resulting factor structures were interpreted on the basis of the factors that accounted for more than 4.0% of the variance for a given data grouping and on the basis of items that had loadings of 0.40 or greater. When taken as a whole, the factor structures present a pattern of results that supports the overall conceptualization of the TEAM model. These results are summarized in Table 1, which shows the percentage of variance accounted for by the factors identified from each of the five training phases.

Table 1

Summary of Factors Identified for
Each Phase of Team Training

TRAINING PHASE	FACTOR IDENTIFICATION		
	TEAMWORK	TASKWORK	TEAM/TASK
BASICS			7.6
PRE-MIDTERM	22.0 5.8	5.5	
MIDTERM	8.0 6.0	8.8	
POST-MIDTERM			26.0 25.8
FINAL			8.4

These data show that the Basics phase of training produced only one substantial factor. This factor loaded most heavily on items related to various aspects of task performance as well as team coordination, cohesion, and communication. It is interpreted as being related to the formation of basic team skills in the earliest stage of training. Two factors were identified in the Pre-midterm and Midterm phases. The first of these is clearly a "teamwork-centered" factor. It consistently loaded on items related to team member's perceptions of activities that involve working with other team members, communication, cooperation, and relationships within the team. The table's double entry for this factor in both the Pre-midterm and Midterm phases indicates that the same factor emerged from both of the analyses of the data groupings that included the data from these two phases. The second factor identified in the Pre-midterm and Midterm phases is identified on the basis of its loadings on items related to the organization and performance of assigned tasks. Items comprising this factor are concerned with efforts to complete the tasks as well as the performance outcomes associated with the tasks. Thus, it is considered to be a task-centered or "taskwork" factor. A single large factor emerged in the final two phases of training. This factor is somewhat different from the one that emerged in the Basics phase. It seems to represent a merger of the two factors identified in the two previous phases. It loads heavily on items associated with both teamwork and taskwork. In

this case, however, these items do not seem to be independent of each other as they were in the prior two phases.

Thus, consistent with the theory underlying the TEAM model, these results indicate that NGFS trainees begin with a wide variety of performance concerns related to the development of team skills. In the second and third phases of training, they express independent concern for teamwork- and taskwork-centered activities. This supports the notion that team members are (a) learning to perform their tasks by discovering the performance rules, exchanging task-related information, learning to operate equipment, etc., while also (b) working to enhance the quality of team interactions by establishing relationships with other team members, developing more efficient patterns of coordination and cooperation, and strengthening team roles, cohesion, etc. Following the Midterm phase, the factors related to these separate kinds of activities merge into a single factor involving both kinds of activities. This suggests that the team has matured to a point where their task- and team-related activities become indistinguishable with respect to their relationship to team performance. In total, these findings provide an initial validation of the model and approach that serve as the basis for this research program.

Data from the TSRQ were also analyzed in order to examine the extent to which the questionnaire items are sensitive to changes in the perceptions of team members across the five phases of training. The results of the separate analyses of variance for each of the 21 items indicated that several of the team-centered and several of the task-centered items yielded significant differences across the phases. Thus, according to the perceptions of the team members, the team's team- and task-related activities are improved as a result of training.

This finding was further explored by examining the data from each TSRQ item separately for groupings of the three most effective teams and the four least effective teams. This division of the teams (into roughly the top one-fourth and bottom one-fourth performers) was made on the basis of performance scores on the Final performance exercise. Again, several of the team-related and several of the task-related items revealed differential patterns for the more effective and less effective teams across the five phases of training.

This result is illustrated in Figure 2, which presents the factor scores for the teamwork and taskwork factors averaged separately for the more effective and less effective teams for each phase of training. These data indicate that in the more effective teams, perceptions about the knowledge of other team members concerning the performance of their assigned duties increased steadily (became more positive) across training. On the other hand, data for the less effective teams indicate that training had a relatively small impact on the perceptions of these teams (particularly the perceptions related to task-related items). In effect, these teams reported that their team members manifested smaller increases in job knowledge as a result of training. The less effective teams did not benefit from training in the way that the more effective teams did. Thus, it seems that additional attention should be devoted to examining the nature of team training received by these teams.

INSTRUCTIONAL PROCESSES

In summary, the TSRQ data indicate that team behaviors change as a function of training and that, at least for the more effective teams, these changes are reflected in increasingly positive perceptions of the team. Apparently, all the teams enter training feeling pretty good about their abilities. However, as the performance of the more effective teams improves, they express more positive team perceptions; this seems to be less true for the less effective teams. Other data not reported here (see 12) also indicate that the teams which do best enter training with more positive attitudes, benefit from more decisive leadership, engage in a higher proportion of more effective team behaviors (and correspondingly fewer ineffective behaviors), and require less intervention from an instructor.

In an attempt to understand why some teams benefited from training more than other teams (apparently with less "instruction"), an effort was made to document the processes employed by instructors and to determine how they dealt with the varying instructional requirements of different teams. Instructor behaviors were examined through direct observation of NGFS training and by conducting interviews with the instructors. Semi-structured interviews were conducted with six of the eight NGFS instructional staff members. The interviews were then transcribed and summary statements were categorized by topic. When possible, instructor comments were further subdivided in terms of their application to the more effective or less effective teams. Based on the observations and the contents of the interviews, a model was developed to describe the NGFS instructional processes (see 16). The primary purpose of this model was to highlight the instructional decisions, strategies, processes, methods, etc. employed in this team training setting.

While a full discussion of the Instructional Processes Model is beyond the scope of this paper, it should be noted the model identifies 10 process-related stages of NGFS training. Each stage involves several instructional processes, some of which are formally required as part of the prescribed training exercises, while others are informally conducted by instructors. In essence, the model indicates that instructors begin by conducting pre-training assessments of the capabilities and training needs of the teams. These assessments are made very quickly and informally (based only on the opinion, experience, and insights of the instructor) during the team's initial briefing. However, based on these assessments, the instructor selects a training approach (formal, informal, socratic, etc.) for use in later stages of instruction.

Teams are categorized by instructors into at least four types based on their assessed

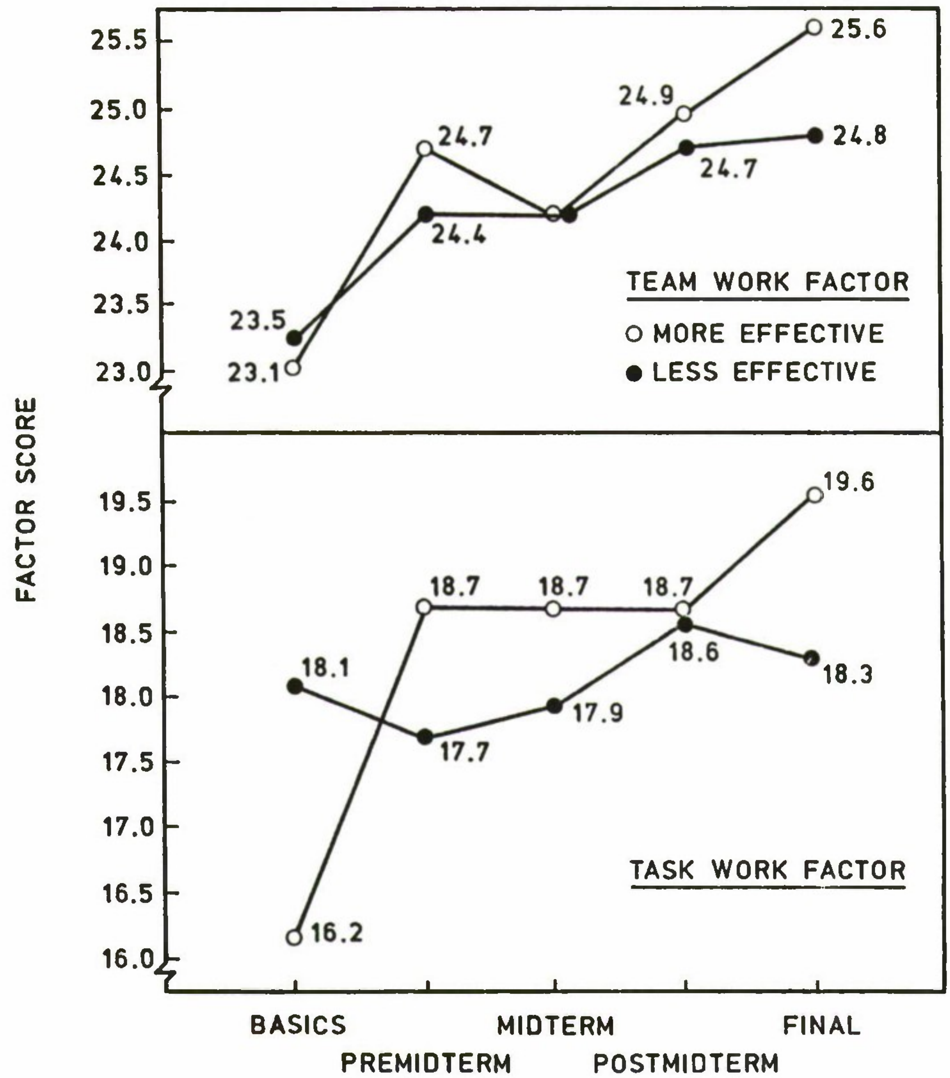


Figure 2. Average Factor Scores for More Effective and Less Effective Teams.

levels of knowledge and attitudes (high and low knowledge crossed with high and low motivation). Considerably different instructional approaches are used in presenting information, guiding the team's performance, and providing feedback to these different types of teams. However, the instructor continuously evaluates the training and adjusts his approach as necessary. Instructors indicate that highly motivated teams are relatively easy to train (although those with low knowledge levels require more time), but that they tend to invest somewhat less effort in the training of teams with low levels of motivation. These teams do not want to be "bothered" with additional information or effort, and instructors are likely to be "turned off" by their lack of motivation.

CONCLUSIONS AND RECOMMENDATIONS

The point here is that on the basis of an informal (probably incomplete and perhaps incorrect) assessment teams are instructed in considerably different ways. Although the system is somewhat self-correcting with respect to instructional approach, it does appear that the teams which could benefit most from team-centered training (those with low levels of motivation) seem to receive less of such training. While this linkage has not yet been firmly established, it can be suggested that the failure of less effective teams to "mature" in terms of the development of teamwork behaviors may be a result of insufficient teamwork training for these teams.

Based on the limited findings to date, it is recommended that formal assessment tools be developed and implemented for use in determining the pre-training levels of task-related skills, teamwork-related capabilities, and motivation and attitudes. In addition, a standardized system should be developed to help the instructor translate the identified levels of abilities and attitudes into clear statements of training needs and approaches. This system should stress teamwork training for teams with low levels of motivation. That is, the system should seek to optimize training approaches on key teamwork variables such as those discussed here. In addition, the training approaches should be standardized so as to provide more formal and consistent feedback to all teams. Other performance aids should also be examined as potential ways to enhance the instructor's ability to assess trainees, monitor critical team behaviors, provide timely feedback, conduct thorough debriefs, etc. Finally, more thorough and formal training sessions should be developed to train instructors to conduct pre-training assessments of trainees, recognize critical team behavior problems, provide appropriate teamwork-centered feedback, etc. The use of videotapes of the performances of effective and ineffective teams might be very useful for such training. While other recommendations are forthcoming from this research program, those identified here should provide an initial basis for substantial enhancements to current team training technology.

REFERENCES

1. Bales, R.F., and Strodtbeck, F.L., "Phases in Group Problem Solving," Journal of Abnormal and Social Psychology, Vol. 46, pp. 485-495, 1951.

2. Bell, M. A., "Phases in Group Problem-Solving," Small Group Behavior, Vol. 13, No. 4, pp. 475-495, 1982.

3. Bennis, W.G., and Shepard, H.A., "A Theory of Group Development," Human Relations, Vol. 9, No. 4, pp. 415-437, 1956.

4. Bennis, W.G., and Shepard, H.A., "A Theory of Group Development," In G.S. Gibbard et al. (Eds.), Analysis of Groups, Jossey Bass Publishers, San Francisco, CA., 1974.

5. Caple, R. B., "The Sequential Stages of Group Development," Small-Group Behavior, Vol. 9, No. 4, pp. 470-477, 1978.

6. Cissna, K.N., "Phases in Group Development: The Negative Evidence," Small Group Behavior, Vol. 15, No. 1, pp. 3-32, 1984.

7. Davis, L. T., Gaddy, C. D., and Turney, J. R., An Approach to Team Skills Training of Nuclear Power Plant Control Room Crews, NUREG/CR-4258GP-R -123022, General Physics Corporation, Columbia, MD, 1985.

8. Denson, R. W., Team Training: Literature Review and Annotated Bibliography, AFHRL-TR-80-40, Wright Patterson Air Force Base, OH.: Logistics and Technical Training Division, Air Force Human Research Laboratory, 1981.

9. Dyer, J., State-of-the-Art Review on Team Training and Performance, Fort Benning, GA.: Army Research Institute Field Unit, 1984.

10. Fisher, B.A., "Decision Emergence: Phases in Group Decision-Making," Speech Monographs, Vol. 37, No. 1, pp. 53-66, 1970.

11. Gersick, C.J.G., "Time and Transition in Work Teams: Towards a New Model of Group Development," Unpublished Manuscript, 1985.

12. Glickman, A. S., Zimmer, S., Guerette, P. J., Montero, R.C., Miller, D.L., Campbell, W. J., Woodard, E. A., Salas, E., & Morgan, B.B., Jr. The Evolution and Maturation of Navy Teams During Team Training, Report No. NAVTRASYSCEN TR-87-016, Naval Training Systems Center, Orlando, FL., In Preparation.

13. Hackman, J.R., and Morris, C.G., "Group Tasks, Group Interaction Process, and Group Performance Effectiveness: A Review and Proposed Integration," In L. Berkowitz (Ed.), Advances in Experimental Social Psychology, Vol. 8, pp. 45-49, Academic Press, New York, 1975.

14. James, L.R., Gustafson, S.B., and Sells, S.B., Final Report: Development of Effective Leaders: The Need to Consider Situational Specificity Verses Cross-Situational Consistency, Contract No. NC0014-80-C-0315, Group Psychology Program: Office of Naval Research, Arlington, VA., 1985.

15. Kennedy, J. L., "The System Approach: Organizational Development," Human Factors, Vol. 4, pp. 25-52, 1962.

16. Miller, D. L., Guerette, P. J., Glickman, A. S., Morgan, B. B., Jr., & Salas, E. Instructional Processes in Team Training, Report No. NAVTRASYSCEN TR-87-017, Naval Training Systems Center, Orlando, FL., In Preparation.

17. Morgan, B.B., Jr., Glickman, A.S., Woodard, E. A., Blaiwes, A. S., & Salas, E. Measurement of Team Behaviors in a Navy Training Environment, Report No. NAVTRASYSCEN TR-86-014, Naval Training Systems Center, Orlando, FL., 1986.

18. Salas, E., Blaiwes, A. R., Reynolds, R. E., Glickman, A.S., and Morgan B. B., Jr., "Teamwork From Team Training: New Directions," Proceedings of the 7th Interservice/Industry Training Equipment Conference and Exhibition, American Defense Preparedness Association, Orlando, FL., 1985.

19. Tuckman, B.W., "Developmental Sequence in Small Groups," Psychological Bulletin,

Vol. 63, pp. 384-399, 1965.

20. Tuckman, B.W., and Jensen, M., "Stages of Small Group Development Revisited," Group and Organizational Studies, Vol. 2, pp. 419-427, 1977.

ABOUT THE AUTHORS

DR. BEN B. MORGAN, JR. is a Professor of Psychology at the University of Central Florida. In the past, he has served as Director of the Center for Applied Psychological Studies and as Chair of the Department of Psychology at Old Domininon University. His major research interests include human performance assessment, the performance effects of stress, the recovery of performance from stress effects, the acquisition of complex skills, and team training and performance.

DR. EDUARDO SALAS is a research psychologist with the Human Factors Division of the Naval Training Systems Center. He is currently serving as Program Manager and Principal Investigator for NTSC's team training and performance program. He has done research on training evaluation, human performance assessment, skill acquisition, personnel psychology, and team training and performance.

DR. ALBERT S. GLICKMAN is an Eminent Professor of Psychology and Head of the Organizational Effectiveness Laboratory at Old Domininon University. His past positions include Vice President, Advanced Research Resources Organization; Deputy Director, American Institute for Research; Chief, Personnel Research Staff, Department of Agriculture; and Director, Psychological Research Department, U. S. Naval Personnel Research Activity. His current research interests are organization and management, the changing composition of the work force, and team training and performance.

THE GREAT DIVIDE

Are 'State of the Art' technologies overshadowing operational training effectiveness in the development and acquisition of training devices in TAF?

TSGT. William J. White, 355TTW/LGMD
A-10 OFT Development Technician Team
DAVIS-MONTHAN AFB. AZ.

ABSTRACT

This paper will discuss the impact that 'State of the Art' technology has on the world of simulation training effectiveness. The complexity of recently developed full mission simulators brings into view the realization that the training device is taking center stage in training systems. The focal point must center on mission and training requirements. More simply put, the trainee/instructor training accomplishments are the measure of an effective training device.

Today's full mission simulators are technical marvels. Acquisition and developmental agencies are getting a product that matches or exceeds the required design criteria. The operational users however, tend to end up with a machine that is often difficult to effectively operate and will not satisfy the need for effective training accomplishment. There is a growing division between operational elements and development/logistics agencies. The training device is moving into the focal point of training systems. As training devices evolve there is an underlying tendency of the training device to become a burden to the training system.

This paper will examine solutions to the over development of training devices. State of the art technology can be used effectively if it is used practically. Operational requirements are not as complete and foolproof as is desired. The user must be included in all phases of development. The United States Military has been in the training business for a long time. Operational units have defined training requirements and identified areas of attention. An experienced pilot knows what he wants out of a flight simulator and all too often this insight is lost in the shuffle or is identified at an inappropriate time. Training devices must be concentrated at the greatest level of effectiveness, the user.

INTRODUCTION

Ask a pilot to give his opinion on what he feels would be the perfect flight simulator and his response will most likely be as follows:

1. Complete fidelity. All cues and sensory responses will be duplicates of those received in an aircraft.

2. The instructor will have visual access to the cockpit at all times.

3. The instructor's operation of the simulator mission will flow smoothly and inputs to the student will be as quick or as slow as the instructor sees fit, with no interruption to the training effort. There should be minimal training required to make the instructor proficient in console operation.

In a training systems environment, a pilot might add that the training device will be delivered with the capability of meeting all mission training requirements and be adaptable to any changes in mission training requirements. There is nothing more frustrating to an operational unit than to receive a new training device that is limited to an emergency and instruments procedures trainer. 'The United States Government paid $ xxx million for this trainer. I can only train the most basic of my mission requirements.', is often heard. It is desirable to have training capabilities echo mission requirements as much as possible with no design tradeoffs impacting these requirements. Training requirements are much too often tailored to the tools available, hence we see a trend of training devices becoming a burden to the system rather than an asset.

Great efforts are made during the development/acquisition process to ensure that all available user activities are included. The resources available to contractors in the area of simulation technology are impressive. Aerospace Systems Division (ASD), Wright Patterson AFB OH, has research and development data on many training device issues. Human Resource Laboratory (HRL), Williams AFB, Az. continually publishes reports on major advances in training device technologies. With the resources available, a training device delivered for use with incomplete or inadequate training accomplishment capabilities is unacceptable.

During the design/development process the focus should ideally remain with the mission requirements of the weapons system. Even issues such as variable missions, or multiple missions should be included in design efforts and tradeoff decisions. Training requirements will then evolve naturally with the weapon system.

As is evident in the title of this paper, a growing divide is present. Technology should never be taken as anything less than a tool. All too often contractors rely on technology for answers. The design and production of training devices is a business. Contractors generally meet or surpass established standards. It is during DOT&E and IOT&E that the user has the opportunity to examine the product, but is during operation that problems or shortcomings become evident. If all requirements are not identified by this time, changes or modifications to the system are hampered by schedule or budget restrictions. During recent acceptance testing of a complex new Operational Flight Trainer (OFT), a front line command pilot, participating in the testing of a new OFT, made the comment that 'our (the user) views mean very little and will have no impact on the capabilities of the OFT at this point in the program'. He was right, criteria had been established and the decision to accept the OFT had been made.

The Army and Navy have taken steps to ensure that the 'man in the seat' is attended to in systems design projects. The Navy has 'HARDMAN' and the Army has 'MANPRINT'. The objective of

these programs is to consider the operator as a part of the system throughout the design process. In earlier systems the operator was often tacked on when the design was nearly complete.

The design of civil and military aircraft has been one of the few areas where human-factors considerations have consistently played a strong role. Major aircraft manufacturers all have competent staffs of human engineering people who worry about visual computer displays, task loads on pilots, seating, and other issues. If this ideology was carried forward into training device design, the tendency to over-develop the device would be neutralized. Having aircraft manufacturers develop a simulator concurrent with the design of an aircraft would concentrate established human factor elements into training device design. The Air Force has not taken the Army's and Navy's approach. In the Air Force, human factor consideration in design efforts have been inclusive to weapon system development for many years. It has become evident that this attitude is not holding true in the training device development process.

In using Army Regulation AR 602-2 (MANPRINT policy), the Army believes that it could do a better job of meeting performance specifications with the soldier in the loop, and thereby reduce the demand for greater soldier talent and training. To relate this to the Air Force, it can be said that putting the pilot in the loop will reduce the demand for extensive operation training of the simulator and the pilot can concentrate on increasing the effectiveness of training without having the added task of dealing with complex instructor operational control procedures. It is difficult to anticipate all the human aspects when viewing a system from only one point of view in its development cycle. More often than not it is from the technological view point that training device development is centered.

The remainder of this paper will discuss the 'Great Divide' and attention will be centered on the 'man in the seat' approach. It is from this perspective that the total process can be most objectively examined.

TRAINERS IN THE TACTICAL AIR FORCE (TAF)

Training devices in the Air Force fleet are not bad machines. The EF-111A OFT is truly a technological marvel. The new generation of trainers are usable tools for training. This paper is not aiming fault or blame in the current acquisition development process. Morever it identifies a trend that is developing in the process that can be detrimental if left unattended. The missions of today's electronic weapon systems are placing an increased need for enhanced aircrew proficiencies at all levels of training. An untrained or incompletely trained mission task can have dire results.

Safety is always the first issue when discussion turns to the justification of flight trainers. The issue that resources must be protected and lives cannot be needlessly lost is cut and dry. Emergency procedures are always big issues in the development process, and is generally a part of the trainer that is not impacted when tradeoffs do occur. At TAF training centers (schoolhouses), the primary mission is to produce safe, trained aircrew members. The mission of these trained aircrews will change in varying degrees as they leave the schoolhouses. Mission and training requirements of front line flying units will dictate mission effectiveness and survivability. If a training device can effectively train aircrews to perform the mission requirements, operational safety will be an inherent trait of the training process.

The A-10 OFT is a good study on the problem of tailored training effectiveness. The A-10 is a air to ground, low altitude weapons system. At the schoolhouse the training device satisfies the safety training requirements of the local unit. However, the front line units have an OFT with no visual system, resulting in marginal weapons delivery training and minimal combat training capabilities for a unit's specific geographical mission area. Essentially the A-10 OFT is limited basically to very effective emergency and instrument flight procedures training. What went wrong? Money, schedule, and tradeoffs played a major part in propagating the resulting deficiencies. These situations were identified during the early phases of development.

LESSONS LEARNED

A look at older simulators will bring to light the simplicity from which simulation development evolved. An examination of the instructor station on an F-4 or F-111 OFT will show items functionally arranged as in actual cockpit systems. The instructor pilot can relate the cockpit system configuration directly to his instructor station. With the advent of graphic display systems as the newest form of instructor station design, the problem of instructor/system integration was realized. On some simulators the graphic representation of cockpit systems is done as an actual depiction of instruments, panels, controls, etc. On newer simulators the representation is displayed as generalized text. For an instructor to control a training session, his plan of training and evaluation process is directed toward the layout of his training console. The A-10 OFT for example, has a display system that follows the console layout of the A-10 cockpit. Control of certain functions require the instructor to call a family of pages, select a page, locate the function on the page and type in an activation command. The flight instrument display page consists of software generated depictions of actual cockpit instrumentation, the display of each console is adequate by itself, however it is difficult to monitor multiple cockpit systems. There are three (3) display screens and only careful mission planning by the instructor will prevent the flow of the mission from becoming uncomfortable and possibly inducing negative training.

The A-10 OFT instructor's station has many capabilities. From basic cross country displays to full blown EW missions and procedures scoring. The A-10 OFT can do many things, but with three display systems and over 250 pages of information it can easily lead to confusion and difficulty in operation. Compare this to simply reaching across the console to activate a discrete button as in older simulators. This is a vivid example of the training device being designed away from the effectiveness of the user. The instructor's training flow is restricted and only concentrated familiarization training and operation techniques will make the training device an effective tool. (see FIG. 1A & B)

When graphic display systems are discussed the talk quickly turns to the software database as the limiting factor in the design and development of cockpit repetition systems that will correctly reflect cockpit instruments graphically. The software required to support a complex graphic

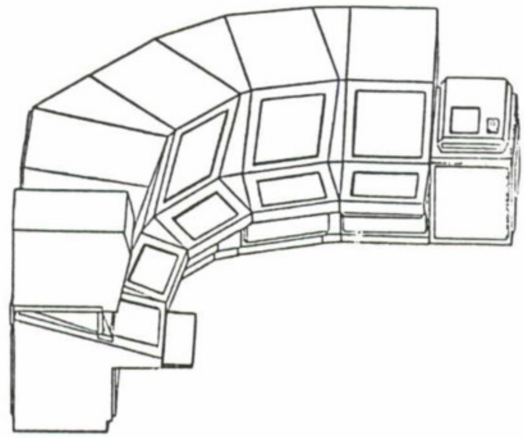


FIGURE 1B.

A-10 OFT INSTRUCTOR STATION

1. Three (3) CRT display system.
2. CRT pages are extensive.
3. Operational control of a training mission is complex.

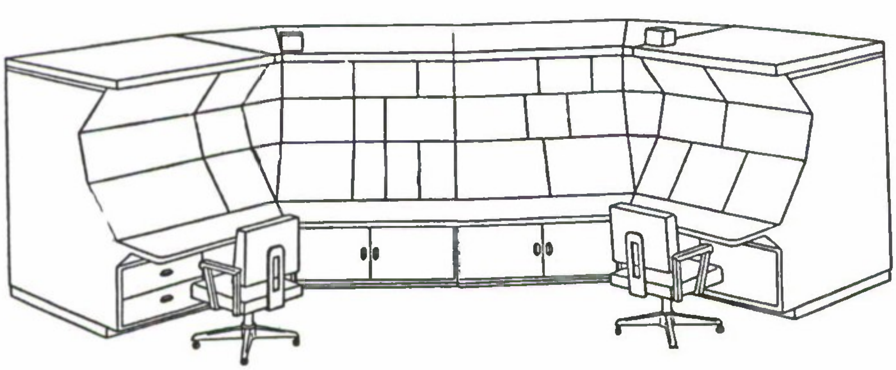


FIGURE 1A.

F-111A OFT INSTRUCTOR STATION.

1. The physical layout of most cockpit systems are echoed on the instructor station.
2. Instructor inputs are simple and mission flow is smooth.
3. Cockpit instruments have some redundancy for maintenence.

TYPICAL DATABASE CONFIGURATION (GENERAL)

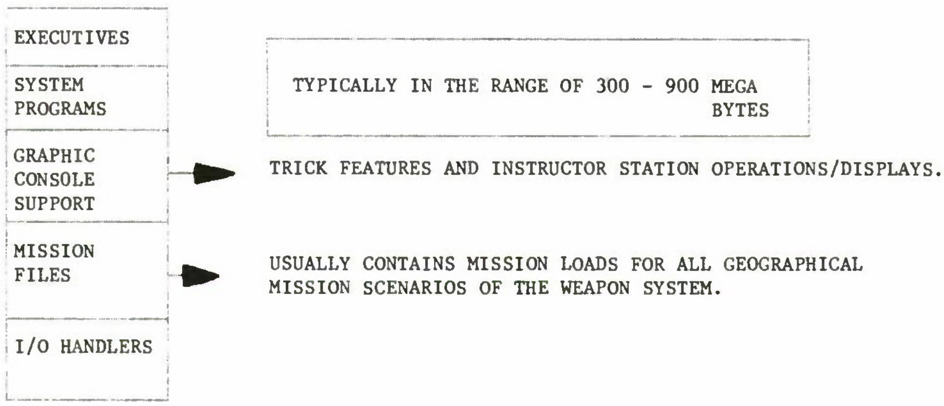


FIGURE 2A.

GP-4B COMPUTER SYSTEM TYPICAL DATABASE CONFIGURATION

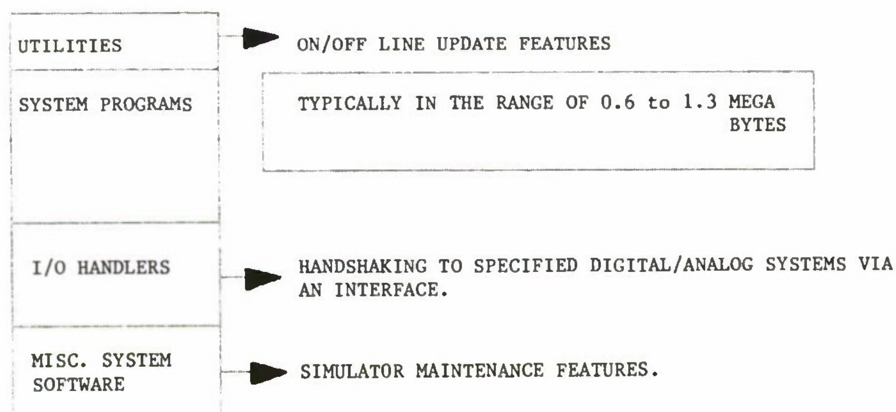


FIGURE 2B.

display system can easily demand 20% of a simulator software database. Text-only display systems allow a reduction in the number of lines of code required to generate a display system, but this technological tradeoff impacts the training device instructor's ability to use the device effectively, by changing his frame of reference from instruments to numerical or text readouts.

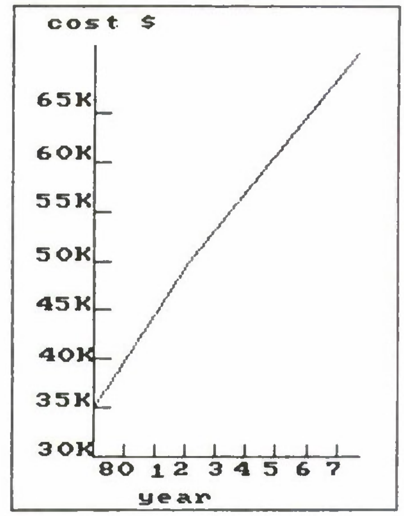
Technology has also given training devices more trick features such as playback/record, procedures scoring, pre-programmed mission scenarios, etc. These features can be justified and used adequately but will remain only as effective as the instructor's ability to control the training device to meet his training requirements.

Research and Development work in the area of instructor station design is moving at a technological rate and not an operational rate. This divide between technology and operation is a problem which again impacts the user. Direction to make the training device a more effective tool centered at the user is imperative.

Software is another big issue. It can confidently be said that the emergence of faster, larger and generally improved computational systems has had the greatest impact on the development of training devices. With the increased computational ability comes the desire to use tbis technological crutch as prominently as possible. Computer manufacturers want to use their systems to the greatest extent possible. Too many times the Air Force expects the diversity of new systems to be as complete as possible. When writing purchase descriptions (P.D.s) and statements of work (S.O.W.s) it seems to be accepted procedure to request the most for the least. Simply put, the Air Force tends to want it all!

The GP-4B computer system used in older F-4 and F-111 simulators can be taken as an example. With 1 megabyte of drum storage and a 1.33 micro sec. execution time, it is considered extremely antiquated in the technological context. The F-4 and F-111 OFTs have been using the GP-4B since the mid 1960's. The computer satisfied established trainer requirements and in the design approach of theses devices the simulation burden was placed greatly on hardware system designs. The instructor stations used actual aircraft instrument repeaters. Analog circuitry controlled a majority of the systems effectively, with minimum computer interfacing.(see FIG. 2) As a result, software support costs and impacts were minimal. A look at the EF-111A OFT brings to light an entirely different situation. The EF-111A OFT utilizes three powerful SEL 32/8760 computational systems. The software database is easily 900 times greater than the previous F-111 OFTs. When an examination of the two trainers is made the improvements of the EF-111A's capabilities over the previous F-111x's capabilities are not radically significant. The instructor station of the EF-111A consists of three RGB displays and most of the cockpit system depiction is in text only. This hampers the instructors ability to use the training device effectively. The EF-111A console is difficult to operate. The trainer does not include a visual nor a motion system. The cockpit environment is limited to instruments, procedures, and aircrew reaction/data management decision evaluation.

The support required to maintain and integrate software updates to the OFT's data base is time



A-10 OFT HORIZONTAL SITUATION INDICATOR (HSI) COSTS

FIGURE 3.

consuming due to the incorporation methods and all the approval agencies that get involved. A simple fix of a software problem can easily take 6 months to be delivered to the user. On the GP-4B changes could be installed at the earliest availability of the simulator for maintenance.

The main point here is that technological improvements have placed a burden on the training system. Will enhanced technological machines satisfy immediate needs in training systems? Stepping back two steps to go forward one should not be viewed as negative progress. The statement, 'newer is better' need not necessarily hold true in all cases of simulation advancement. Civilian industry must work side by side with the Air Force to ensure thorough performance and design evaluations. In a 1979 staff summary, Maj. Gen. Carey (then USAFTAWC/CC) stated that the Air Force tends to generalize requirements to the point that contractor design efforts are impacted as expanded or newly realized requirements surface. Eight years have passed and this situation is still prominent.

STIMULATION vs. SIMULATION

The limits placed on the expandability and supportability of training devices are greatly impacted when special purpose non-aircraft components are used in trainer design. One example of this is the use of simulated flight instrumentation in the A-10 OFT. The availability and quantity of these instruments has been a sore spot. A few years into the OFT's life, the support cost of these special purpose units rose and this cost was passed on to the Air Force. (see FIG. 3) Unnecessary OFT down time for awaiting parts was incurred, while these units go back to the original manufacturer for repair.

A look at the F-111F OFT will show a large percentage of actual aircraft instrumentation. In using the stimulation technique, money, time, and training hours were saved. An inherent redundancy for many cockpit systems was realized, and more often than not, that redundancy saved OFT down time over the life cycle of the trainer.

Update and modification of aircraft systems becomes easier and more timely when stimulation

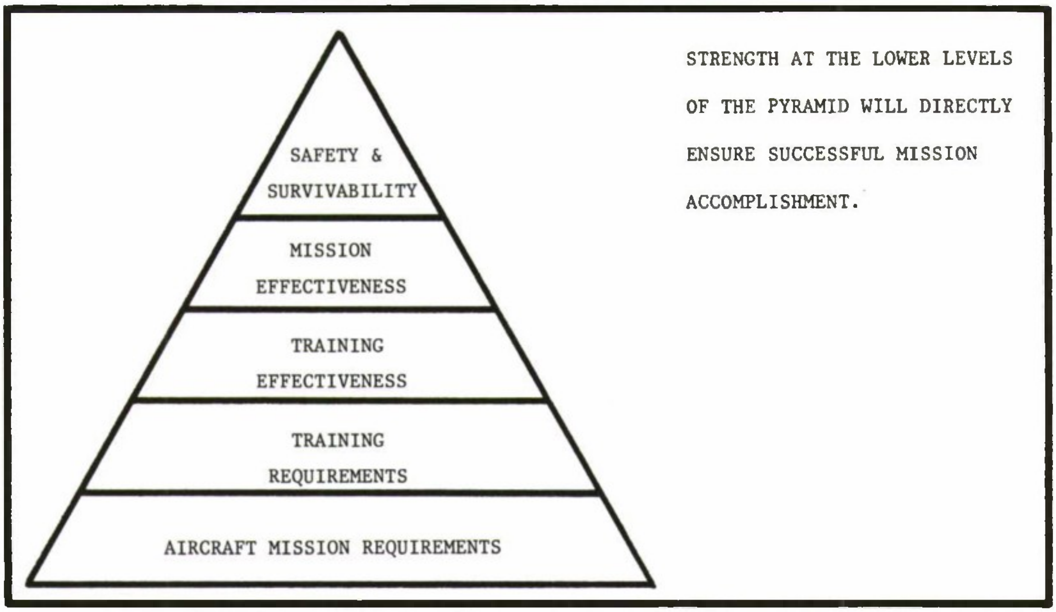


FIGURE 4.

techniques are used in lieu of simulated indicators. It is much easier to update an indicator with minimal stimulation software changes, than to redesign a specialized instrument and reconstruct the simulation software. The ancient problems of time, money and fidelity are attended to in an efficient manner when aircraft modifications are as simple as possible to incorporate into a training device.

The simulation vs. stimulation debate could go on for days. One fact stands clear, the update capabilities of a stimulated system saves time, life cycle support costs, and helps in the drive for increased fidelity.

TECHNOLOGY ON THE MOVE

For some reason analog systems are regarded as artifacts of an old technology. Digital is the buzz word of the new generation of trainers. Technology has advanced forward in leaps and bounds in the area of analog systems, as well as digital. Analog systems provide the immediate benefit of being minimally dependant on software. Instruments and control loading systems operate more smoothly when driven by analog controllers. With the head-aches realized in the area of managing and controlling the ever increasing size of software databases, the use of advanced analog systems integrated with state of the art digital systems will be a welcome and effective relief.

Digital Radar Landmass (DRLMS) is another study in technological goldplating. The training capabilities of DRLMS are extensive, but the drawbacks of the system include cost, software database size and integration. In the most basic example, do all DRLMS equipped trainers utilize the capabilities of the system? Do DRLMS equipped trainers need all of the capabilities of DRLMS? The decision to utilize DRLMS was made as a possible answer to the need for a generic device in the acquisition/development of new generation training devices. The tricolor and grayscale systems utilized on older simulators had drawbacks directly related to technological inadequacies and supportability. The performance of these systems satisfied the training requirements when the system was operationally sound.

The DRLMS does meet its established training requirements. In general terms it is an overkill. Projection systems are cheaper, the required hardware is half that of a DRLMS, and the software required to run the system is limited to hand shaking and attitude control with the host computational system. The projection system is basically a technologically upgraded Analog Radar Landmass System (ARLMS) which can perform to required standards. LANTIRN type radar projection systems are available and are as geographically complete as needed. This means that AF photo/recon data or Defense Mapping Agency data can be updated in the projection type radar system as needed in an expedient manner. This is but a small example of simplicity in design and efficiency in costs. The DRLMS on the other hand is (can be) delivered with an immense geographical data base. Changing geographical mission areas is relatively time consuming and update to the data base requires either digital map building or a concentrated software program update.

As far as instructor/student interface is concerned, tactical aircraft training devices do not have the convenience of having instructor(s), figuratively watching over the shoulder, as do cargo and larger type training devices (i.e. B-52, C-130, and KC-10). The effectiveness an instructor has in a tactical simulator is in direct relation to the effectiveness with which the instructor can control the training mission. As stated previously, design efforts in the instructor's operational capabilities have made mission accomplishment in the training device more complex than is needed. Stimulation, functional instructor station design, performance capabilities, and thoroughly reviewed training requirements are the suggested solutions to immediate training device technological directions, or misdirections.

THE NEXT STEP

The 'Great Divide' need not occur when training systems are developed. As long as the focal point of requirements and criteria information is constant and visible, industry will deliver complete and effective training

devices. The user is the key body when the final lauds and degradations of the training device is assessed. The Air Force has had a history of inconsistency in developing and validating specified requirements and effectiveness directions.

Simulation in TAF is taken extremely seriously. Training effectiveness and training requirements have been in critical focus for a number of years. A system must be developed that will allow for the complete integration of all existing and predicted mission and requirement deviations. User input at the unit level is often overlooked due to generic or established standards of operation.

Aircraft cockpit design has always been directed toward its most prominent factor, the pilot. If training devices were designed with the same emphasis as aircraft, the user students and instructors will have a major impact on design, function, and operational capability of training devices. While the student emphasis in simulation design is generally adequate, it is the complete system that lends itself to redirection. The Air Force is falling into the hole being opened in the "Great Divide". If the link between acquisition/development and operational performance requirements is to be unified, it is at the Air Force level that emphasis must be made to bridge that gap.

One solution to the situation is to define areas of expertise and concern, allow maximum interface between all concerned organizations, and allow industry to play a bigger role in the early stages of development. A working group with defined and full time administrative organization would be a step in the right direction. The emphasis on mission requirements, training requirements, and regular evaluations of training effectiveness must be a primary goal when a training device is the issue. (see FIG. 4)

Industry can also assist in the gap closing process. Sympathy toward simplification and diversification of all functional operation characteristics of a training device is one way to control technological runaway. Conferences such as the Interservice/Industry Training Systems Conference, are valuable interface mediums. Industry must utilize Air Force simulation agencies and research efforts as greatly as possible. Industrial interface cannot be limited to technical areas. It is not always possible for industry to work hand in hand with operational components for inputs to training issues. This is where a dedicated working group could be most beneficial. (see FIG. 5)

A meeting was held on 2 October 1979 between Gen. Stansbury, AFSC DCS for Procurement, and vice presidents of Boeing, American Airlines, G.E., Reflectone, Singer-Link and Goodyear Aerospace at the request of Gen. Slay to discuss industry's view of Air Force aircrew training device acquisition. This meeting identified many key issues:

a. AF doesn't use Integrated System Development (ISD) process so it doesn't know what it wants or how to get it.

b. AF doesn't accomplish smart trade-off studies.

c. Poor overall Air Force planning for simulators.

d. AF goldplates requirements and then backs down, sometimes too far.

e. Simulator contractors need help dealing with aircraft prime contractors.

f. AF would be smarter to not always demand mil-spec compliance.

g. AF demands excessive data.

h. AF has no champion of simulators.

Take into account that these points were addressed eight years ago. The Air Force has made steps to redirect simulator acquisition. ISD has been used more effectively, as is seen in the B-52 WST. Trade-off studies are more complete. Requirement goldplating is still an issue and steps to reduce it are evident in newer acquisition efforts. With all this, the poor overall planning issue is still a subjective concern.

As a result of the aforementioned meeting, the Air Force assessed its policy on simulators and the resultant staff summary report sent to the Air Force Chief of Staff (15 OCT. 1979), stated as follows;

> "A review of Air Force activity in procurement and employment of simulators for aircrew training reveals a major effort in progress aimed at taking advantage of improvements in technology - most notably in the area of sensors (visual, motion, and radar). This technology holds promise of full mission training capability for simulators -a step beyond contemporary trainers."

Technology is visibly a major factor in Air Force training device development. The emphasis on technology was established years ago with good reason. Sensory/cues/tasking technologies are key figures in the drive for aircraft/simulator fidelity. The problem arises when the question of "how much fidelity do we need from a flight simulator?" is addressed. The research and development costs of fidelity achievement are relatively large in relation to total simulator design. It is easy to justify research and development efforts. A problem arises when the relative costs of R&D impact funding for new training devices. Something has to give and it usually includes trade-off decisions involving the basic training device. The Air Force does not help itself in this area, when it asks for everything and goldplates requirements. Take the A-10 OFT for example. A visual system was required from the development stages of the acquisition. The Air Force asked for a technologically state of the art visual for the F-15, F-16 and A-10 trainers. This was called Project 2360. The cost of this visual was enormous in relation to the total cost of the training devices. When funding became restrictive the project was shelved. As a result the A-10 to this day has single window VITAL IV systems on two of thirteen trainers. The Air Force placed extensive capability and training requirements on the requested visual; however the A.F. wrote itself into a corner. There were cheaper visual systems available that would have met basic mission training requirements. But the Air Force stuck with the idea of a cosmic, primarily air to air visual system for all three types of trainers, rather than purchase a more adequate, more available, better suited visual for each type device. In the current acquisition process, it is difficult and almost not accepted procedure to

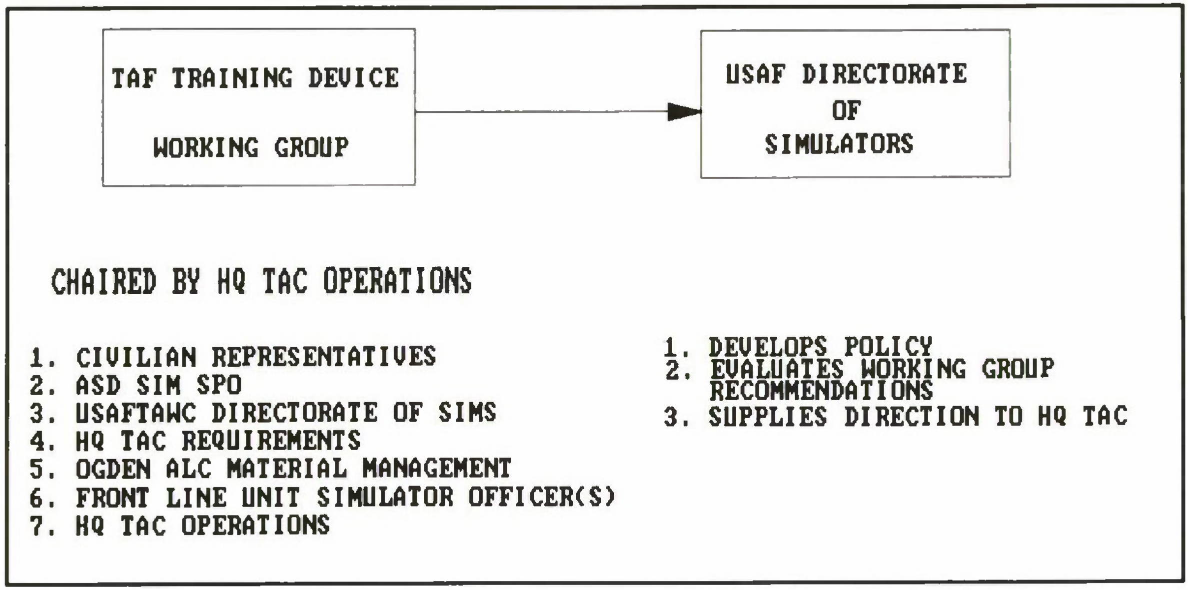


FIGURE 5

WORKING GROUP MEMBERSHIP AND FLOW

change the Statement of Need (SON) to allow for a deviation in requirements in order to obtain a simpler system. As a result the A-10 OFT has no visual system on 11 trainers. Being that the aircraft is an air to ground, low altitude "tank killer", without a visual the OFT is relegated to the role of a very good procedures trainer. Planned flexibility is the key. Working with what is available and designing to mission effectiveness requirements are directions needed to ensure that effective training devices are included in training system environments.

SUMMARY

This paper indicates the importance of thorough interfacing between users, acquisition agencies and industry. Training effectiveness was examined as a product of technological simplicity and instructor control capabilities. The growing "divide" was explained to be an information integration problem between acquisition/development agencies and operational components of TAF. The basic problem was mentioned to center around the availability and use of state of the art technologies in the design of training devices and its impact on the user when delivered for use. The operational capabilities and functionality of design was shown to be the second leg of the division, and an overlooked tendency of acquisition agencies in the development effort. Technological simplicity was related to older training device design and current simulation design efforts. It was shown that functional capabilities differed in operational and mission training effectiveness as state of the art devices evolve.

Solutions to the problems were given in the technical and administrative forums. Technically, a direction of stimulation, simplicity and functional design was examined. Administratively, it was suggested that a "champion of simulators" be established through the use of a dedicated agency comprised of all facets of simulation (industry, users, AF MAJCOM's).

All inputs to training system issues must be examined at all levels and phases of development. Since training devices are important parts of training systems, it is important to address the issue of burdening the system. Design and requirement efforts must be concentrated at the student/instructor level with minimal impacts to the training effectiveness process. Let's put training back into the hands of the user and assure the highest level of safety, survivability, and mission effectiveness.

ACKNOWLEDGEMENTS

The support of the following people made this paper possible: Col. Joseph Crozier, and Lt. Col. Lowell Pate of the UNITED STATES AIR FORCE TACTICAL AIR WARFARE CENTER, SMSgt Harold Senff, MSgt William Nunn, TSgt Charles Cramer, TSgt Alan Zero, SSgt Kerry Morse, and SSgt Richard Esquivel, members of the A-10 OFT Development Technician Team, and Mr. Ralph Nelson, Lead Software Engineer, A-10 OFT Developmental Engineering Prototype Site.

REFERENCES

Courtice, Andrew J., Capt "THE NEXT STEP . . . , Emerging Issues and Implications for Management in the Acquisition of Training Systems in the Air Force.", Paper presented at the 8th Interservice/Industry Training Systems Conference, Salt Lake City, November 1986, pp. 97-105.

Horowitz, Scott J., Capt., PhD. "Measurements of Transport Delays In A State-Of-The-Art F-16 Flight Simulator", Paper presented at the 8th Interservice/Industry Training Systems Conference, Salt Lake City, November 1986, pp. 316-321.

Stonge, James R. "The Role Of Evaluation In Training Systems Acquisition", Paper presented at the 8th Interservice/ Industry Training Systems Conference, Salt Lake City, November 1986, pp. 375-384.

"TOO MUCH TOO SOON; Information Overload.", IEEE SPECTRUM, Vol. 24, Number 6 (June 1987), pp. 51-55.

USAFTAWC, "Simulator Day - Flash report #5: A consolidation of simulator issue meetings and policy directions of the Air Force.", 5 October 1979.

ABOUT THE AUTHOR

TSgt William J. White has been a member of the A-10 OFT Development Technician Team, Davis-Monthan AFB AZ., since Nov. 1986. Prior to that time he was a Digital/Navigation Tactics Training Device technician on the F-111E, F-111F, and A-10A Flight Simulators.

DESIGN OF A GENERIC TRAINING DEVICE CONTROL CONSOLE USING ADA

V. Faconti
L. B. McDonald, Ph.D.
Harris Government Support Systems Division
Winter Park, Florida

ABSTRACT

Several factors set the stage for control console designers who wish to compete in today's training environment. Chief among these are various DoD initiatives to reduce the costs and increase effectiveness of training systems. The DoD mandate to use Ada* is a good example. This paper documents a program of research aimed at developing a design approach to realize the DoD cost-effectiveness goals in the training device control console area. This approach features increased use of modular generic sofware solutions which can be applied over a wide range of situations. At the same time, the approach allows for modification to accommodate specific requirements as needed. A functional baseline was developed based upon reported console design studies and then expanded through developmental testing and user surveys. User reactions and Ada lessons learned are also discussed.

INTRODUCTION

Over the past few years, the cost of simulation hardware has plummeted while the cost of software has skyrocketed. While much of this hardware cost reduction is due to advances in miniaturization, the dominant effect has been the result of modularization of hardware. Instead of designing and building dedicated hardware for specific applications, engineers select modules that will perform the required functions. In order to fulfill the varying functional requirements, these hardware modules must have a built-in flexibility such as programmable functions or expansion capabilities that can be implemented by the end user without hardware modification. Hardware modules with this flexibility are used in a number of applications, thus spreading the development cost over a large number of units and lowering unit cost.

To achieve lower costs in software, the training systems industry must develop flexible modular software packages that can be used repeatedly in a number of training system projects, thereby, distributing development costs over a number of applications. This emphasis on reusable training system software modules is in keeping with the larger DoD Ada Initiative for weapon systems in general.

In order for hardware manufacturers to develop modules that will be used repeatedly, they had to first determine what functions must be performed repeatedly. This same function analysis is required by software developers in order to determine what software functions will be used repeatedly in training systems. The U.S. Air Force has performed an analysis of flight simulator functions and has developed the following list of simulation (functional) modules for a weapon systems trainer (as defined in the SOW for Modular Simulation Design.):

Aerodynamics	Visual
Flight Controls	Navigation
Flight Station	Support
Instructional System	Electronic Combat
Motion	Radar
Propulsion	Weapons

In order to be flexible enough to be used repeatedly on separate flight simulators, these simulation software modules must consist of a large number of packages that will carry out the various functions likely to be needed in the diverse applications of the future.

In this paper we will concentrate on the module labeled by the Air Force as Instructional System. To achieve the needed flexibility for the module, we must first determine the functions the module must perform in the future. Since the primary purpose of the instructional system is to support the simulator instructor/operator, these functions can be derived by doing an analysis of instructor/operator functions.

BACKGROUND

The imposition of Ada fully supports a generic instructor console concept, and further, even simplifies its implementation. Figure 1 illustrates the advantages realized by reusability of Ada software. Since Ada software is designed to compile and execute on any Ada-compatible processor, reusable instructional software source code need only be developed once. Each subsequent implementation requires only compilation on the target processor and the writing of driver packages to meet the specific needs of the instructor and interface hardware.

*Ada is a registered trademark of the U.S. Government, Ada Joint Program Office.

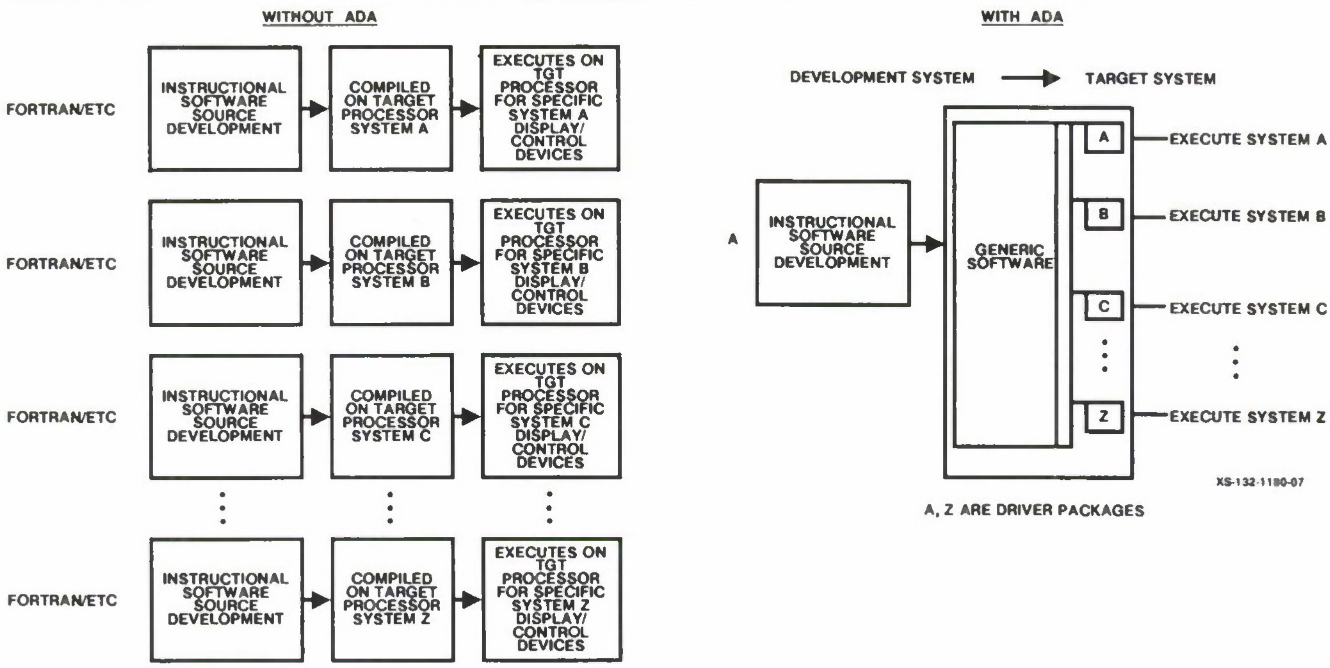


Figure 1. Advantage of Ada

Accordingly, we are now presented the tools needed to explore the concept of a Generic Instructor/Operator System (IOS). The combination of microprocessor power/cost, advanced raster or flat panel display technology and Ada Offers this opportunity.

As a more pragmatic understanding of training technology has evolved over the years, so has an understanding of the instructor's role in a simulation training environment, and with that, a clearer understanding of the functions that are concomitant with this role. Much of the material discussed below provides an understanding of the functions performed by current and future instructors. The premise of this work is based on an understanding that the role of the instructor station is to provide a mechanism by which the instructor can view and alter the training problem.

Many desirable features of control consoles have been well documented in the literature, several are listed in Table 1. Figure 2 is a sequential flow of instructor/operator functions and how they interact with the other modules and functions of a flight simulator. Generic IOS software must perform the functions labeled as IOS functions in order to be reusable on a large number of flight simulators as other classes of simulation training devices.

The design approach followed here was driven by a desire to capitalize on technology, and at the same time meet the needs of a great number of training situations. Therefore, modularity became a central performance requirement. With the performance aims understood, the next step was to lay out the functional basis for building the system. The functions of a generic control console should be common to a great many control situations.

Table 1.
Summary of Desirable Control Console Features

INSTRUCTIONAL FEATURES	LISTED BY (SEE REFERENCES)
RECORD/PLAYBACK	1, 4, 7
REMOTE REPEATER DISPLAY	1
HARDCOPY	1, 4
MANUAL FREEZE	1,4
AUTOMATIC FREEZE	1,4
PARAMETER FREEZE	1, 4
DEMONSTRATION	1, 4
DEMONSTRATION PREP	1
AUTOMATIC MALFUNCTION FAULT INSERTION	1, 4, 7
AUTOMATIC MALFUNCTION INSERTION EXERCISE PREP	1, 4
INITIALIZE FUNCTION	2, 3, 5, 7
PERFORMANCE EVALUATION FUNCTION	2, 3, 7
DEBRIEF STUDENT FUNCTION	2, 3, 7
DATA MANAGEMENT FUNCTION	2, 3, 5, 7

XS-132-1181-07

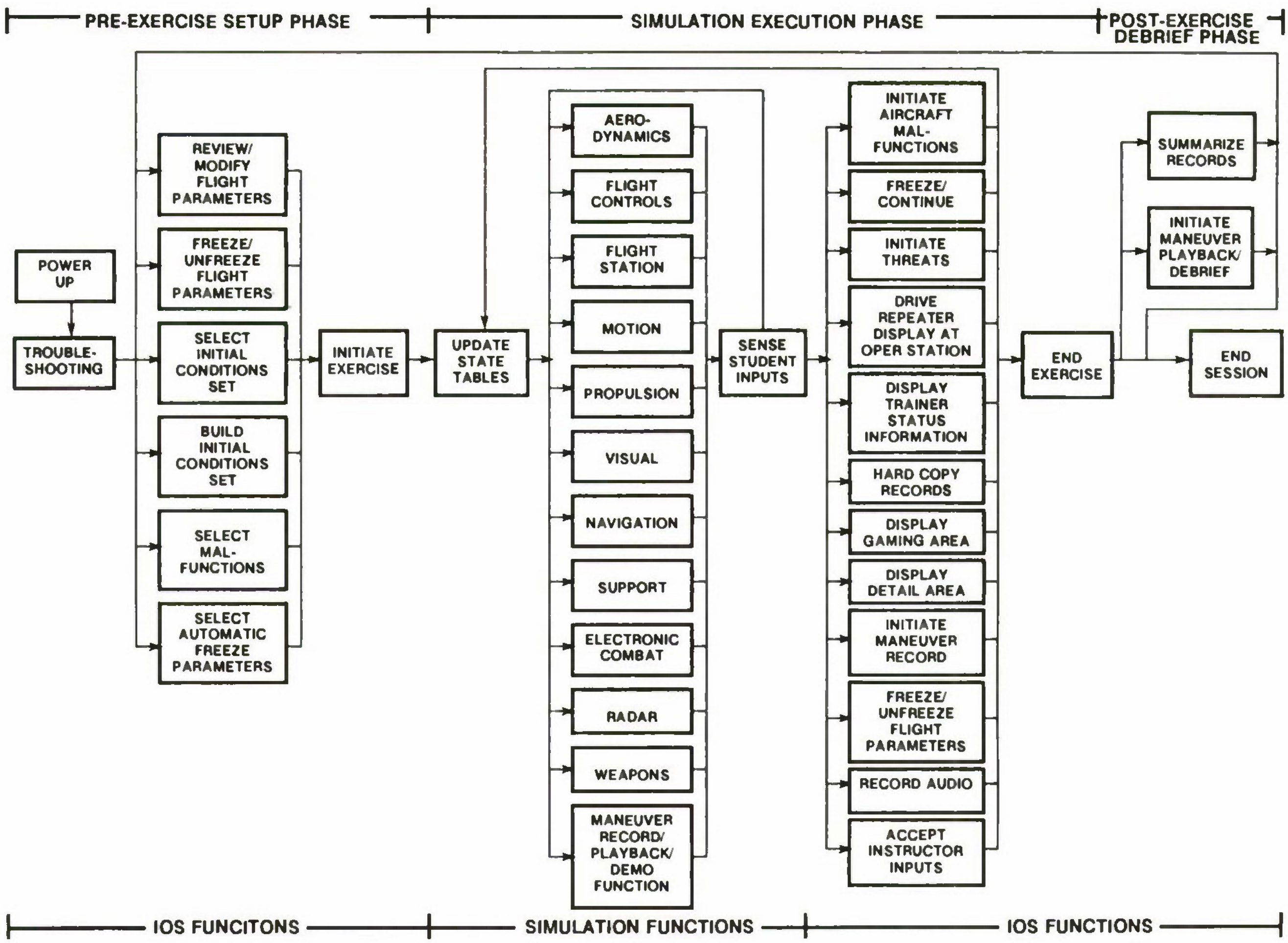


Figure 2. Required Generic IOS Functions XP-076-855-07-2

DESIGN MODEL FOR A GENERIC IOS

While the functions in Figure 2 are all different, many of the underlying software tasks are the same. Consequently, the list of functions can be distilled down to the following elements.

1. Display data to the instructor in textual format. This function can be used to not only display data resident in the simulation data pool but also to display data contained in mass memory for elements such as initial conditions, mission scenarios, environmental data sets, navigation data sets, tactical data sets, etc.

2. Display data to the instructor in symbolic format. There are several variations of this kind of a display, e.g., navigation maps, tactical maps, GCA, formation flying, weapons loading, etc. However, this task is essentially reduced to mapping state data pertinent to the student's position into a symbolic view of the problem.

3. Accept data singularly from the instructor to alter the instantaneous state of the simulation. This function can be used to execute initial conditions, insertion of malfunctions, reset training scenarios, alter specific symbol values and other basic functions in the system.

4. Accept data in blocks to redefine the problem. This function can be thought of to serve several different features listed in Table 1. Initial condition data, environmental data sets, reset, record/playback and demonstrations are a few examples.

5. Store data in blocks for later retrieval. Again, this function can serve several of the features listed in the reference Table. In particular, scenario generation functions can be performed by this type of a generic element. In addition, storing data for record/playback, demonstration and initialization is also performed by this element.

6. Perform mathematical functions on simulated data. This capability would be used to display massaged data to an instructor either on a CRT format or via some hard copy mechanism.

Table 2 lists the required IOS functions and indicates which software shell element satisfies that requirement.

Table 2.
Feature Comparison

INSTRUCTIONAL FEATURES	SHELL ELEMENTS 1	2	3	4	5	6
RECORD/PLAYBACK				X		
REMOTE REPEATER DISPLAY	X	X				
HARDCOPY	X	X				
MANUAL FREEZE			X			
AUTOMATIC FREEZE			X			
PARAMETER FREEZE			X			
DEMONSTRATION				X		
DEMONSTRATION PREP					X	
AUTOMATIC MALFUNCTION FAULT INSERTION			X			
AUTOMATIC MALFUNCTION INSERTION EXERCISE PREP			X			
INITIALIZE FUNCTION				X		
PERFORMANCE EVALUATION FUNCTION						X
DEBRIEF STUDENT FUNCTION	X	X				X
DATA MANAGEMENT FUNCTION	X		X			

XS-132-1182-07

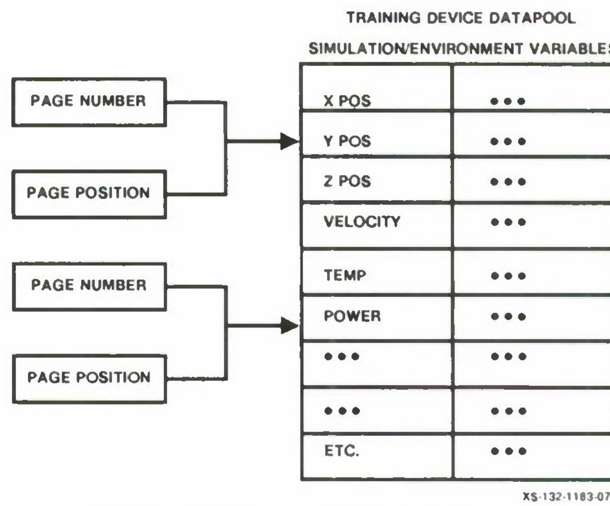


Figure 3. Display and Modification of Datapool Variables

Implementation Considerations

One of the key issues briefly mentioned earlier which allows this approach to be fully achievable is the use of a data driven system. This means that all interaction from the instructor to the simulation problem can occur through data pool variables (in the traditional sense). Thus, a textual page is simply a collection of ASCII characters with pointers to variables in the simulation datapool. Alteration of these variables occurs through some mechanism (be it keyboard, touchscreen, mouse, voice, etc.) mapped to that page. A data driven approach allows the instructor to identify data he wishes to modify and to actually modify that data. This concept is illustrated schematically in Figure 3. A similar analogy can be drawn for activation of mission segments, for example. By using a data driven methodology to index into mission data sets, a textual page can be used in the same manner as mentioned above to identify the data set to be recalled from mass storage as illustrated in Figure 3.

Organizing the instructional function in a training device in this manner clearly supports the object oriented definition required for design using the Ada programming language. Each generic element in this system can function as an object, with other elements of hardware also serving as objects in the design of the overall object model.

The following discussion addresses a generic model and how it is defined in order to implement the instructional features needed in a training device. A typical model is illustrated in Figure 4. All elements except the simulation state block are part of the generic process. This presentation shows a single input device which takes some action on the simulator state by, for example, changing the value of the altitude. Similarly, it could affect the instructional state itself by activating a new display image on either of the two displays shown or activating the store/retrieve data blocks shown. To further illustrate, consider the block marked "Store data blocks". This would handle storing data for record/playback, storing data for CRT page usage, storing data for missions, etc.

As shown in Figure 5, the block from the previous illustration can be broken down further into sub-blocks, each performing a generic function. For example, block #1 is responsible for storing sequential data at a prescribed frequency rate onto some mass storage device. For record/playback and demonstration the frequency rate would be a number greater than 1. However, when used for initialization and reset, the frequency would be set to 1 and one block of data would be stored. Block 2 is a CRT page index for textual pages, while block 3 is the block for symbolic or graphic pages. The last block shown is the training scenario system where scenario design is structured in hierarchical sets. In a typical case, these sets would consist of a set of initial conditions, a set of environmental conditions, a set of navigation aids required to support the problem, and a set of automated features needed such as malfunction insertion, procedures monitoring, etc. The next level in the hierarchy then would be a definition of those sets to be used. This flexibility in design allows the users to customize the software package to fit their needs and makes the package reusable in a large number of applications.

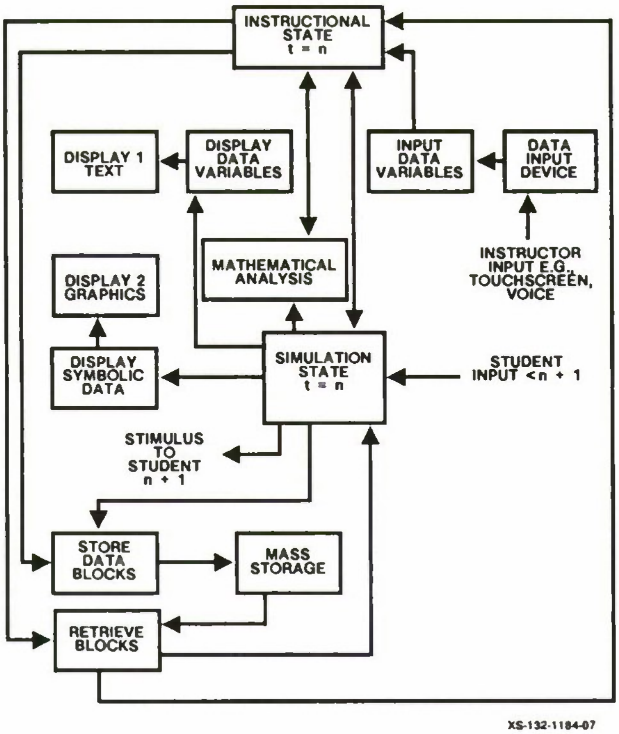


Figure 4. High Level Simulation Model

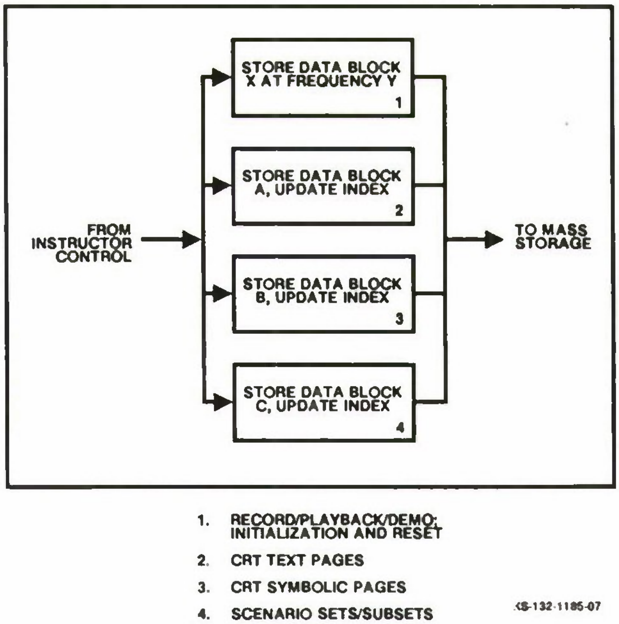


Figure 5. Detail of Store Data Block

Feature Methodology

The implementation methodologies required by this approach are very flexible in that, by adhering to a strict data driven approach on many of the functions, variations of those functions are simple to implement. A few examples will be presented to illustrate this point. Consider first, textual CRT pages, pages with descriptive text identifying variables within the simulation problem as well as field location on a display screen. A data-driven approach requires that textual pages are subdivided into two pieces similar to Singer-Link's ASPT approach. First would be a fixed portion ASCII string which would contain all of the non-changing information on the screen such as variable identification, units and other similar items.

The second part is an update table which is maintained whenever a particular page is activated, and this data is output to the screen surface on some cyclic basis, for example, twice/second. This feature can be easily implemented by constructing the desired page layouts in an off-line mode. The pages are constructed by using the actual fixed textual information required in the proper screen locations and by utilizing a method to identify the data fields where simulation parameters are to be displayed through the use of either the defined symbol dictionary name or some superset of that name tailored for the user population. In addition, by defining action codes located on the screen, it is possible to implement a coding mechanism by defining variables in the datapool which activate many functions. For example, in one case a symbol dictionary location might have an action code which directs, when selected, that a numeric keyboard input will provide data to be inserted into the simulation problem in a certain location in memory. Alternatively, on a different screen or on the same screen for that matter, a location could have the action code, that when activated, adds 1 to the page number being displayed. The result of this would be the selection of a new CRT page.

This approach can be extended to scenario generation by first defining a series of subsets consisting of, as mentioned, initialization sets, environmental conditions sets, etc. The structure of the scenario can be created by collecting these sets into segments, usually numerically coded. Therefore, the same operators that are used for creation of CRT pages may also be used for creation of scenarios with the exception of their storage structure.

Examples of Model Application

The discussion to this point has, of course, been theoretical and abstract. However, this concept has been implemented recently as part of an R&D project conducted by the Harris Corporation in conjunction with the Naval Training Systems Center. The system model was based on the experimenter/operator system (EOS) currently used in conjunction with the Visual Technology Research Simulator (VTRS) located at Naval Training Systems Center. The VTRS itself served as a test bed for evaluation of not only this abstract concept of developing an instructor control system, but also the implementation of a microprocessor-based modular instructor console programmed using Ada.

The system recently developed in Ada is shown in Figure 6 and consists of a series of Ada tasks and packages that are invoked by a series of events. There are four events that control the execution of the software programs, namely, a timer which activates tasks on an iterative basis, the touchscreen, change data from the simulation, and a keyboard. Once execution occurs, the processing tasks being cued by the event tasks would then output information to one of three output elements. One task, of course, is to modify data for CRT pages themselves, either alphanumeric or graphic. Another task would be to provide hard copy of CRT page data and the third task is to transmit data to the simulator to alter its particular state.

Most of what is shown in Figure 6 is generic in the sense that it can be used on any training simulator which follows certain precepts. That is, a simulation data pool or similar repository for current data must be available and a CRT system is used for control. It must be pointed out that, of course, the graphics are unique to the hardware used except in the structure of the model. Only one task would have to be replaced, and that is the task that deals directly with the input/output to the graphics processors. In fact, the entire model is based on this approach and there are, of course, certain graphic depictions which are unique to this training device. However, the methodology for implementation allows insertion and removal of different tasks fairly simply. This technique was also demonstrated during the current research project.

Figure 7 is an example of the application in Ada of the concept of a generic task. As mentioned previously, the function of the CRT page and related control hardware is to provide information to the instructor and further to allow the instructor to insert information into the simulation problem by creating an online page editor. The page created simply defines data associated with a particular function and the generic display task can instantiate any type of page. As can be seen from the figure, all actions associated with the page are defined in the generic task. Thus, by creating a specific and unique data base for each function, a variety of display pages can be created.

In an effort to make the software as flexible and reusable as possible, the display task was designed such that the user can create and modify screens of data (called frames) without re-compiling the code. To create a display frame, the user will select the CREATE FRAME option from a menu, respond to the prompt and name the new frame. The user will then see the Edit Field menu in Figure 8. From this menu the user can insert, delete, copy and move fields on the frame as well as create and delete other frames. If the user selects INSERT FIELD, the Insert Field menu in Figure 9 will appear. To monitor a certain value in the simulator datapool,

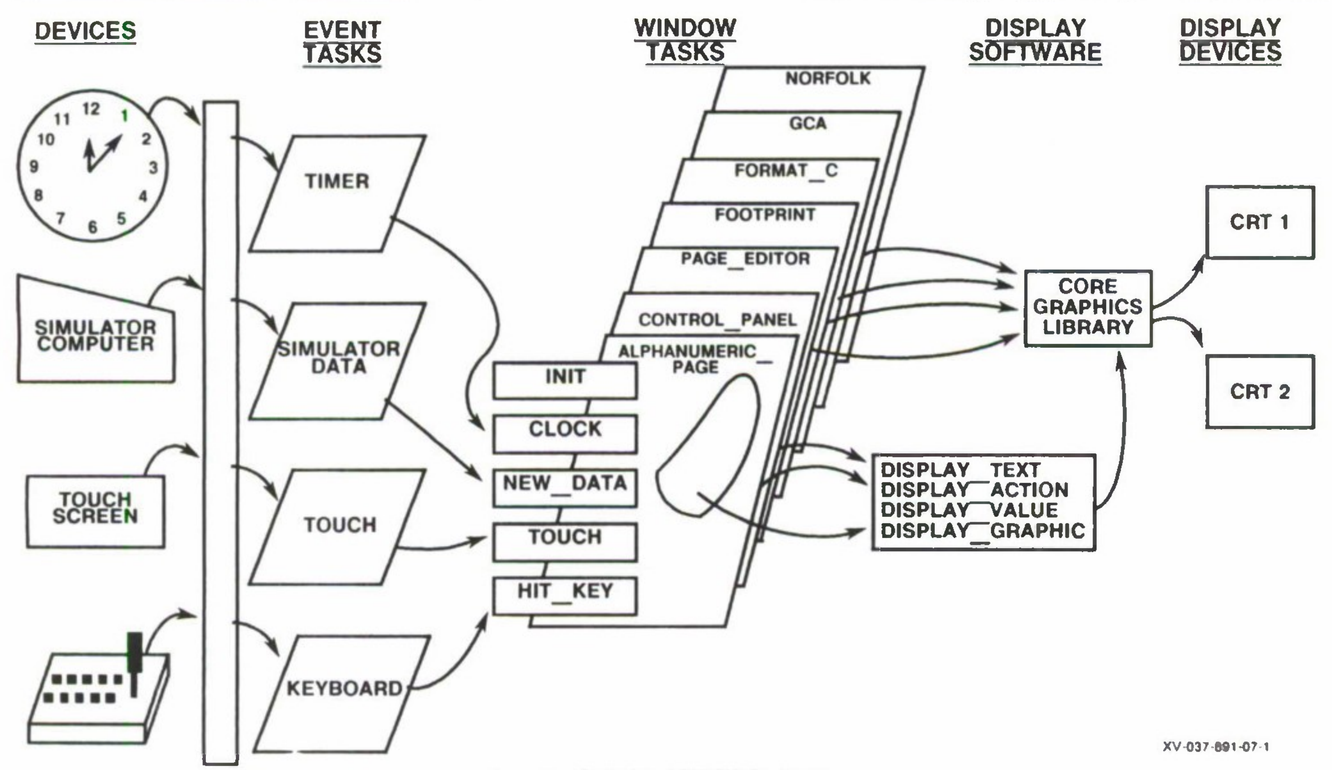
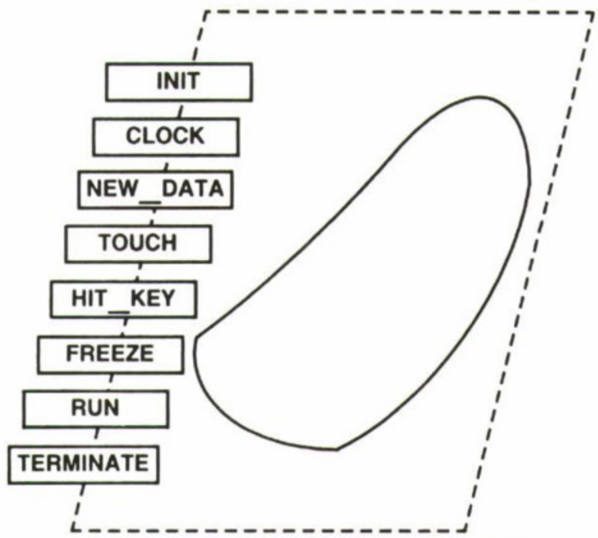


Figure 6. Current MODIOS Software



XP-037-887-07

Figure 7. A Generic Display Task

such as altitude, the user will touch the appropriate square and see a prompt to enter the datapool variable name. A prompt will then appear to touch the desired location. The user will touch the desired location on the screen and the value for the desired parameter will be displayed at that location whenever that frame is displayed. The user will then touch the INSERT TEXT option, insert via keyboard the desired text and touch the desired location. To insert a control function, such as NEXT PAGE, the user will touch the INSERT TOUCH SCREEN option, designate the location on the frame, select the desired action (GO TO FRAME X) from a menu and name the desired frame. With this level of flexibility in frame design the reusability of the software is assured.

LESSONS LEARNED

The work discussed in this paper was conducted as an IR&D project by the Harris Corporation and utilized the facilities of the Visual Technology Research Simulator (VTRS) at NTSC facilities in Orlando, Florida.

In the pursuit of completing the overall design and the resulting implementation, many lessons were learned. These included a much deeper appreciation for the power of Ada and more insight into the design of a generic instructor station console. Throughout the design and development, emphasis was placed on the use of Ada and Ada design methodologies. Results were gathered throughout the project and more specifically during a user evaluation. The following paragraphs summarize these findings.

Data were gathered during three phases of the project.

a. Software Development;

b. Hardware/Software Integration;

MODIOS EDITOR

● EDIT FIELD MENU

- ☐ INSERT FIELD
- ☐ DELETE FIELD
- ☐ MOVE FIELD
- ☐ COPY FIELD
- ☐ MODIFY FIELD
- ☐ CREATE FRAME
- ☐ DELETE FRAME
- ☐ NEXT FRAME
- ☐ PREVIOUS FRAME
- ☐ EXIT EDITOR
- ☐ SAVE AND EXIT
- ☐ SAVE AND CONTINUE

XS-132-1186-07

Figure 8. Screen Editor Options Menu

MODIOS EDITOR

● INSERT FIELD MENU

- ☐ INSERT DATAPOOL VARIABLE
- ☐ INSERT TOUCHSCREEN
- ☐ INSERT TEXT
- ☐ SAVE AND CONTINUE
- ☐ CONTINUE

XS-132-1187-07

Figure 9. Insert Field Options Menu

c. User reaction to the ability of the design to effectively train and operate the Visual Technology Research Simulator (VTRS).

The purpose of the evaluation was to develop data to validate the instructor console station concept, and provide feedback for system improvement. The scope of the evaluation was established by a list of twenty-two evaluation questions which covered both general and specific areas of investigation. Participants in the evaluation, during the software development and hardware/software integration phases, included both Harris and Naval Training Systems Center technical team members.

Software Development and Ada

Findings with respect to Ada as an implementation tool are consistent with those being reported with other Ada projects. Ada is a robust language that offers many capabilities that did not

previously exist in many high level languages. Since an object oriented design approach was utilized on this project, project personnel had to learn Ada as well as how to design with object definitions in mind. Ada provided a direct application of the design in which objects were mapped easily into a programming solution. A very strong front end definition was obtained. The major finding with respect to Ada is that the training of Ada and Ada design methodologies is less painful than originally expected. Additionally, the software integration took approximately two weeks to complete. This quick integration process can be directly attributed to the front end definition and the level of abstraction that was attainable with the use of the Ada language. Debug time was minimal and strongly aided with an effective symbolic debugger. The use of generics expedited the addition of program units. A new task that consisted of a color driven graphics display was designed, coded and tested in less than one week. This was achievable due to the highly structured code, as well as reusable software packages designed into the framework of the instructor console software.

Hardware/Software Integration

The software system was developed on a SUN 3/160C computer system utilizing the Verdix Ada compiler. The system offered an excellent tool in which to do software development. It has the power of the Unix operating system, and the processing power to comfortably support four to five software engineers. This system served as both the development platform and the target run-time system. The conclusion reached is that the Sun workstation offered an excellent development environment but could not provide the processing power required to achieve the iteration rate required for real-time operations in the configuration used. Table 3 summarizes the results of the execution speeds. The slow updates are attributed to the Unix operating system and how input/output is now implemented by the operating system. Bypassing Unix I/O drivers and directly addressing the graphics processor should speed up the through-put rate to an acceptable level. From all indications, given a different I/O interface, Ada is suitable for real-time applications.

Table 3. Execution Times

MEASUREMENT	TIME (MSECS)
SYSTEM TIME CALL	.383
GRAPHIC WINDOW TASKS:	
ALPHANUMERICS	
DISPLAY A PAGE	2208.000
DISPLAY MAIN MENU	2141.000
CONTROL PANEL	
CLOCK UPDATE (ENTIRE) (0:00:00)	83.000
FORMAT C DATA TABLE	
ENTIRE DISPLAY UPDATE	1588.000
GCA GRAPHIC	
ENTIRE DISPLAY UPDATE	449.000
GRAPHICS UPDATE ONLY	422.000
MAP GRAPHIC	
ENTIRE DISPLAY UDATE	199.000
GRAPHICS UPDATE ONLY	189.000

XS-132-1188-07

User Reaction

User reaction was outlined by utilizing ten Marine Corps helicopter and fixed wing pilots. A short training scenario was used to expose each instructor pilot to the generic console. After each pilot completed an exercise, detailed questions were answered. User reaction to the use of the generic instructor console was very positive. The operator console was used to control a training mission involving the SH60B helicopter simulator. A limited selection of features including: call up of initial conditions, in-flight store and recall, change of instructor controlled parameters, hierarchical menus, and the page editor were implemented successfully. The generic instructor console station did adapt to the control needs of the VTRS. Specifically, the instructor pilot's reaction can be summarized as follows:

a. The touchscreen approach with many control features selectable by touching the face of the display screen is an efficient and effective way to control the training problem. The instructor pilots particularly liked the way in which the in-flight store feature was implemented.

b. All subjects noted that the use of color for CRT displays added to the readability and presentation of the training mission parameters.

c. No subject experienced difficulty in console operations. Four of the subjects remarked that the console was easier to operate than those that they had used before. The menu driven approach allowed access to all information within a hierarchical structure.

d. All subjects were able to use the instructor console station within a 15 minute training and orientation period.

CONCLUSION

The results of this work lead the authors to two basic conclusions. First, the concept of a generic IOS is certainly feasible. Hardware technology has reached a level where common modules can be structured to implement all requirements in an instruction console, particularly if designers maintain open architecture such as VME, multi-BUS, PC, BUS, STD bus, etc. Further, by designing the software using a data driven methodology, major elements of the instructional software itself can be used in a variety of trainers.

The second conclusion deals with the use of Ada. Our experience has been that Ada allows you to reach an operation state much quicker than was previously experienced. The problem of servicing I/O must be overcome and from indications

among the commercial software developers this is happening. Adding new tasks to an Ada system is as advertised, i.e., simple and easy, thus lending more credence to the generic IOS concept.

BIBLIOGRAPHY

1. Caro, Paul W., Pohlmann, Lawrence D., and Isley, Robert N. (1979) Development of Simulator Instructional Feature Design Guides, Seville Technical Report TR 79-12, Seville: Pensacola, Florida 32505

2. Charles, John P. (1983) Device 2F119 (EA-6B) WST Instructor Console Review, Technical Report: NAVTRAEQUIPCEN 81-M-1083-1, Naval Training Systems Center, Orlando, Florida 32813

3. Charles John P. (1984) Design Guidelines for Trainer Instructor/Operator Stations, Technical Report: NAVTRASYSCEN 83-C-0087-1, Naval Training Systems Center, Orlando, Florida 32813

4. Faconti, V., Mortimer, C. P. L., Simpson, D. W. (1970) Automated Instruction and Performance Monitoring in Flight Simulator Training, AFHRL-TR-69-29, Air Force Human Resources Laboratory, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio

5. Stark, E. A., Faconti, V., Mortimer, C. (1971) Study to Determine the Requirements for an Experimental Training Simulation System, NATRADEVCEN 69-C-0207-1, Naval Training Systems Center, Orlando, Florida 32813

6. Charles, John P., (1983) Device 2E6(ACMS) Air Combat Maneuvering Simulator Instructor Console Review, Technical Report: NAVTRAEQUIPCEN 82-M-0767-1, Naval Training Systems Center, Orlando, Florida 32813

7. Osborne, S. R., et al. (1983) Three Reviews of the Instructional Support System (ISS) Concept, Technical Report: NATRAEQUPCEN 81-C-0081-1, Naval Training Systems Center, Orlando, Florida 32817

ABOUT THE AUTHORS

MR. VICTOR FACONTI is the Manager of Advanced Engineering, Harris-GSSD/Orlando. He is responsible for system analysis and design for the simulation product line, and has technical and financial responsibility for operation IR&D program, direct support to all Business Development activities and system design for ongoing projects. Mr. Faconti has over twenty years experience in the design of flight simulation systems. He was technical champion in the area of instructor stations and features design. He holds a BS (1964) and MS (1968) from Adelphi University and and has several patents and publications in the area of flight simulation system design.

DR. BRUCE MCDONALD received his Ph.D. in Industrial Engineering from Texas A&M University. As a Program Engineer with Harris Corporation, he manages programs in the advanced Engineering Division. Dr. McDonald has extensive research and applications experience in training device concept formation, design and evaluation as well as user-computer interaction.

A RESEARCH TOOL TO IMPROVE THE EFFECTIVENESS OF PERFORMANCE MEASUREMENT WITHIN THE IOS

Amanda M. Williams, Ph.D.
Tactical and Training Systems Division
Logicon, Inc.
San Diego, CA

Gary S. Thomas, Ph.D.
Air Force Human Resources Laboratory
Operations Training Division
Williams Air Force Base, AZ

ABSTRACT

Functions of the Instructor/Operator Station (IOS) include the display of information necessary for the instructor to monitor and assess student performance and to provide the student with diagnostic feedback. To support these functions, reliable, valid, and useful measures of student performance are necessary along with graphic capability to display relevant information.

The Air Combat Maneuvering Performance Measurement System (ACM PMS) is a prototype research device developed to address monitoring and debriefing requirements of the IOS. The ACM PMS includes state-of-the-art graphics display capabilities and traditional and innovative measurement algorithms to support ACM training.

The device has been interfaced with the Simulator for Air-to-Air Combat (SAAC) and the Air Combat Maneuvering Instrumentation (ACMI) range and is capable of collecting, displaying, storing, analyzing, and replaying ACM performance information gathered from training exercises conducted in both the simulator and on the range. The co-location of the SAAC and the ACMI provides a readymade environment for ACM operational training research. With the implementation of the ACM PMS, automated data collection from both simulator and airborne ACM training is possible.

The ACM PMS was designed to support a program of research intended to develop, refine, and validate useful measures of performance and to develop ways of presenting this information to both the instructor and the student. High resolution, real-time, interactive graphics are expected to yield innovative approaches to providing measures of student progress and to supplement and replace traditional methods of debriefing.

The paper describes the ACM PMS development to satisfy SAAC and ACMI user requirements, the system's capabilities, and plans to use the device for measurement validation and performance monitoring and debriefing research.

INTRODUCTION

The instructor/operator station (IOS) constitutes the interface between the instructor and the flight simulator and, in most cases, between the instructor and the student. The functionality of the IOS directly affects the quality of instruction that the student receives. Thus IOS features, specifically designed to meet instructor's needs, facilitate training. This, in turn, leads to more efficient use of the training device and other training resources.

The Air Force is conducting a program of research intended to result in future procurement of IOSs that are better designed and more cost efficient than prior systems. Warner[1] and Charles[2] have produced design guidelines that specify human factors and training functional requirements for the IOS. As military specifications, the documents will lead to the procurement of IOSs that more effectively and efficiently support simulator training.

Instructional support features which allow the instructor to control, monitor, instruct, and evaluate training exercises are expensive components of the IOS, but also facilitate simulator use as a mor effective aircrew training device (ATD). In a series of surveys (Polzella[3]; Polzella[4]; Polzella and Hubbard[5]; Polzella and Hubbard[6]) problems were identified in the specification, implementation, and use of instructional support features in a large sample of Air Force ATDs. Based on these findings and independent surveys and interviews, Easter, Kryway, Olson, Peters, Slemon and Obermeyer[7] developed the Instructor Support Feature guidelines for application to IOS design.

These studies have identified performance measurement as one of the least understood and accepted of the instructional features. In addition to taking on a great variety of implementation configurations, automated measurement algorithms have been implemented prior to the conduct of rigorous validation studies. Accurate, well understood automated performance measures must be provided to support instructors in their evaluation of student progress. Furthermore, graphic capability to display relevant performance information must be available. These IOS functions are necessary for the instructor to monitor and assess student performance and to provide the student with diagnostic feedback. It should be stressed, however, that performance measurement should support, not replace, the instructor evaluation process.

AIR COMBAT PERFORMANCE MEASUREMENT

Formally validated performance measures are not available for tactical air combat. Although some measures have been developed and have received operational aircrew acceptance, rigorous validation studies have not been carried out. Operational acceptance of a measure establishes some degree of validity for the measure. However, formal evaluation in the form of establishing the relevance, reliability, and freedom from contamination has never been demonstrated in the tactical arena.

The development and formal validation of accurate and objective measures are necessary for the evaluation of student performance and for accurate feedback. In addition, they provide measures of the effectiveness of training devices. Differences in airborne measures of student performance taken before and after simulator training can be used to quantify the effectiveness of the training device. Such measures allow training designers to adjust training syllabi to optimize the use of flight simulators and other training resources including instructors and aircraft. Used in this way, the measures facilitate the management of resources within a training program.

Air combat is perhaps the most demanding type of flying. Increased maneuverability of modern aircraft and the presence of human computer interfaces in the cockpit have increased taskload demands. The high rates and conditions of uncertainty in which motor responses, perceptual responses, and decisions must be made puts the air combat pilot at the limits of human performance.

Airborne performance has eluded study for many years, because psychologists have had to rely on aircrew debrief. Although adequate for operational debrief, verbal and written recollections cannot completely recreate the ACM event with scientific accuracy. Therefore, the collection of data on airborne behavior was the major obstacle to the psychological study of air combat and flying tasks in general.

Accurately and reliably measuring air combat performance is a goal of training psychologists. With the introduction of instrumented ranges such as the Air Combat Maneuvering Instrumentation Range/Tactical Aircrew Combat Training System (ACMI/TACTS), airborne events could be accurately recreated, stored, and graphically replayed. The ranges for the first time provided excellent

Figure 1. The ACM PMS is a graphics-based workstation and data base designed to support air combat performance measurement research.

capabilities for real-time airborne data collection. These objective data, when combined with audio replay and aircrew interview, allow psychologists to recreate the airborne events with the objectivity and precision necessary for objective studies in performance measurement development.

Flight simulators represent a useful testbed for the development and validation of air combat performance measurement. Data are typically available in flight simulators that describe relative position of opposing aircraft, pilot maneuvering of aircraft, energy management, and weapons effects. These data represent necessary ingredients of algorithms needed to describe air combat performance of proficient pilots. A research program with the specific goal of developing and validating air combat performance measures is described below.

AIR COMBAT MANEUVERING PERFORMANCE MEASUREMENT SYSTEM

Overview

As a first step in developing and validating performance measures, the Air Combat Maneuvering Performance Measurement System (ACM PMS*) was developed to collect both simulator and airborne training data. The co-location of an ACMI and the Simulator for Air-to-Air Combat (SAAC) at Luke AFB provided a readymade environment for operational air combat research. The ACM PMS has been developed as a research tool to study automated performance measurement in addition to other IOS features in support of training in air combat. The integration of the ACM PMS with the SAAC and ACMI enhances the research opportunities in this environment by providing convenient data collection from both devices. New concepts for displays and graphic replays can be tested, and performance measurement algorithms can be developed and modified.

Design and Development Process

The ACM PMS was developed through an interactive process, working with the instructors at the SAAC and at the ACMI. The layout of the displays, the performance information to be displayed and the performance measures resulted from a series of interviews and discussions with the operational training personnel. The design of the system to provide research capabilities for performance measures, displays and other features was based on the operational design.

Functional Description

The ACM PMS (Figure 1) is a graphics-based workstation and data base designed to support both research and operational training activity in an air combat simulator as well as actual airborne engagements. Data are collected during training events to provide dynamic, real-time graphics displays and graphic replay of the events at a later time. High-resolution graphics display the training engagement as it unfolds, providing both out-of-the-cockpit and rotatable three-dimensional views of the air combat engagement. In addition, real-time performance measurements of aircraft relative position and specific excess energy are computed and displayed during the training engagement. All data are available for replay for research purposes or for operational debriefing.

In addition to all information available during the training event, relative position data and specific excess energy are plotted over time during replay. These time history curves help to show the progression and relative advantages of engaging aircraft over the course of the engagement. Other playback features include arbitrary positioning to any point within the recorded engagement, with pursuant playback at slow, normal, or fast.

In addition to the graphics display capabilities, the user interface consists of an interactive, touch sensitive menu for selection of displays, data base operations, annotation of data, and flagging specific events. Flags are available to enter event-related data such as tactical radio calls that are not machine detectable. A feature is also available to standardize start and stop times of ACM engagement recordings. This feature is included to ensure the accuracy of time referenced performance measurements.

Simulator for Air-to-Air Combat (SAAC)

The SAAC is a dual cockpit air combat flight simulator with a full-field visual system. F-15 and F-16 cockpits are available at each station. The visual system is composed of eight large CRT displays covering the canopy area of each cockpit. Engagements flown in the SAAC can include up to three "aircraft." Two of the aircraft are flown from cockpits. The third is flown by the simulation computer or by an instructor using rudimentary flight controls at the IOS.

* The ACM PMS was developed by Vreuls Research Corporation (VRC) and Logicon, Inc. under contract to the Air Force Human Resources Laboratory. The authors wish to acknowledge the contributions of Dr. Wayne Waag of AFHRL/OT, Lt.Col. Bart Raspotnik (ret.) of the Simulator for Air-to-Air Combat at Luke AFB, the VRC project members led by Richard Obermayer and the Logicon project members led by William Comstock.

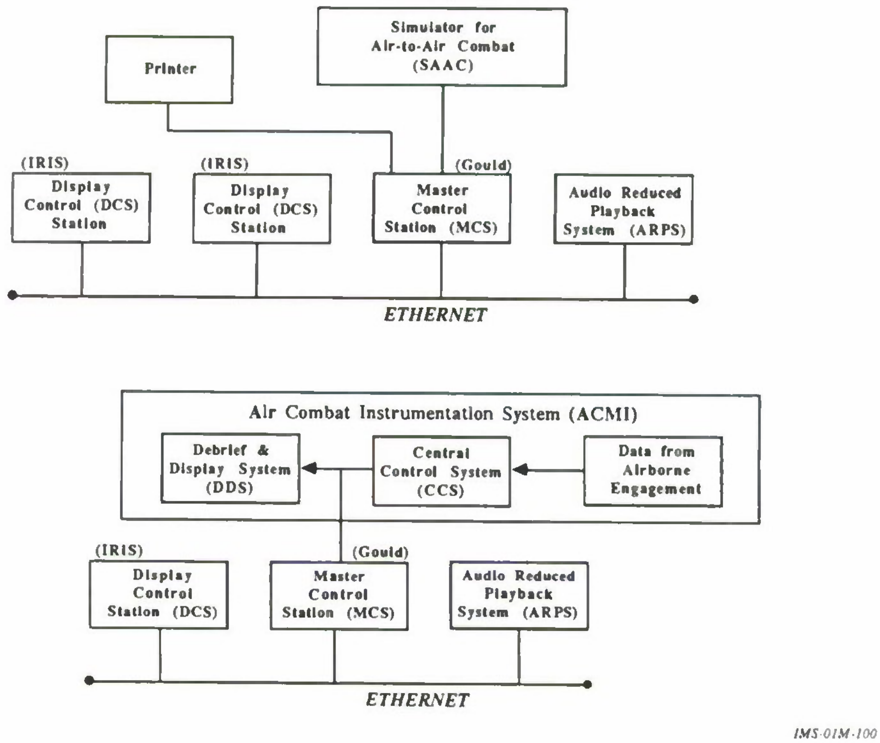


Figure 2. (a) In the upper figure, the ACM PMS configuration at the Simulator for Air-to-Air Combat is shown. (b) The ACM PMS configuration at the ACMI is shown in the lower figure.

Two ACM PMS high-resolution, graphics workstations are available at the SAAC. They are the Display Control Stations indicated in Figure 2a. One of these stations is located at the IOS and the other is placed in a debriefing area. The two stations can operate independently. Graphic replay of an engagement for debriefing or research purposes can be run on the remote station at the same time a live training engagement is being recorded and displayed by the station at the IOS. The Master Control Station is the direct interface to the SAAC. It computes the performance measures and accesses a large relational data base. The Audio Reduced Playback System provides digitized recording of all spoken communications during the training events.

Air Combat Maneuvering Instrumentation (ACMI) Range

ACMI ranges provide a realistic environment for airborne ACM training. The ACMI ranges are approximately 40 miles square. During ACM training tracking stations on the range receive information from on-board computer pods. By triangulation, the ACMI computers determine the location of the aircraft taking part in the ACM engagement. Up to 16 aircraft may be tracked, with eight aircraft considered to be a high activity level. Missile launches are simulated, and the success or failure of each shot is indicated. All computations of position, missile launch and graphics representations are performed in (near) real time and stored on tape. This information is used to graphically depict, in real time, the ACM engagement. Training officers can observe the ACM engagement as it occurs and replay it for debriefing after the training engagement.

One ACM PMS workstation is provided at the ACMI. It is the Display Control Station indicated in Figure 2b. As in the SAAC configuration, the Master Control Station is the direct interface to the ACMI. The ACMI is composed of several major subsystems, and the location of the interface to the ACM PMS is indicated. With the exception that it has one Display Control Station, all components of this configuration of the ACM PMS function exactly as they do in the SAAC configuration.

Data Configuration

The data collected by the ACM PMS are stored and used in three different file structures according to the function they are to serve. Data are initially recorded and stored in a sequentially organized file which is used to support the graphic and audio replay of air combat engagements. This data file contains all information needed to completely restructure and replay the engagement.

Data to be used for research are reduced from the replay file to a large relational data base. All data in an engagement are accessible by crossreference to all other data within an engagement. The data base includes such logical relations as aircraft and inter-aircraft position information for every aircraft pair, weapon effectiveness information, aircraft control inputs, including throttle, speed brake and stick position, information with respect to radar and lock-on procedures, and calculations of the candidate performance measures. These data are collected continuosly throughout an engagement.

The third form that the data takes is as input to statistical packages run off-line on an IBM PC/AT. Subsets of data are selected from the data base and formatted and transferred to the AT through an RS232 interface. The data are then analyzed on the IBM PC/AT using the Statistical Package for the Social Sciences (SPSS).

Candidate Performance Measures

Two candidate performance measures that have received user acceptance were included in the system design. The ACM PMS records data, computes algorithms and displays results for Energy Management and the All Aspect Maneuvering Index (AAMI).

Energy Management is based on the concept of specific excess energy. It indicates how well a pilot manages the potential and kinetic energy of the aircraft. Pruitt, Moroney and Lau[8] described the development of the energy management display and recommended a formal evaluation of the instructional effectiveness of the displayed information.

The concept of energy management has been used in the instruction of 1 v 1 basic fighter maneuvering (BFM) at the Fighter Weapons School at NAS Miramar for several years. An Energy Management Display was implemented on the Navy's TACTS range at NAS Miramar in 1977 and has been successfully used in the instruction and debrief of ACM engagements since that time. Operational acceptance, and therefore validity, of the concept has thus been demonstrated.

The AAMI represents interaircraft position and relative offensive state. It is derived from the Readiness Estimation System (Oberle and Naron[9]; McGuinness, Bouwman and Puig[10]). RES is a comprehensive system providing a Maneuver Conversion Model, a Weapon Firing Sequence, and a Performance Index (PI). The AAMI is directly related to the PI. The AAMI is a measure of position of the fighter with respect to an adversary aircraft and with

respect to weapons envelopes. An AAMI score is computed for each type of weapon the fighter has on-board. AAMI values range from zero, indicating no opportunity for a shot, to 100, indicating an optimal firing opportunity.

Energy Management and the AAMI have both been used in operational settings and receive some degree of acceptance by operational aircrews. In addition, these two measures compliment each other. During ACM the pilot is constantly evaluating trade-offs of energy state and offensive/defensive position. Before making a maneuver to increase an offensive state, the pilot must evaluate whether the aircraft has enough energy to complete the maneuver. On the other hand, before making a maneuver to increase energy state, position must be evaluated. Combining measures of energy and position therefore give a more complete picture of air combat maneuvering. Displays of these measures are continuously available both during the training engagements on the SAAC and the ACMI and during the graphic replay for debriefing and research review.

PLANNED RESEARCH

Initial research efforts will focus on modification and validation of candidate performance measures. Further efforts will provide measures to describe successful air combat performance, to evaluate pilot and training system performance, and to provide diagnostic performance feedback to the instructor and student.

The effort will begin with the development of a model of air combat performance, against which the performance measures will be evaluated. The measures will be examined for relevance to the working definition, reliability over time and changing conditions, and susceptibility to contamination. An expert systems approach may be incorporated to elicit and model air combat pilot decision-making behavior. Data analyses will identify the major factors in air combat performance. Multiple regression analyses will identify the best predictors of performance.

After performance measures are developed and validated, research will be conducted on how to optimally display this information to the instructors to assist monitoring of student performance across training exercises. Current ACM PMS graphic displays will be modified and upgraded, and changes will be systematically evaluated. Methods of displaying performance information for mission debriefing will also be investigated. Since all relevant student performance is recorded in ACM PMS, instructors will not be required to rely on memory to recreate training events. Research can focus on providing instructors with rapid access to relevant information for instructional purposes. Procedures for supplementing traditional debriefing sessions will be investigated.

Validated performance measures open the door for related research activities in air combat training. Improvements in performance as a result of training in the simulator or on the range can be empirically documented. Measures of performance enable research in such areas as the transfer of training from the simulator to airborne events and the appropriate use of the simulator for remediation of airborne flying problems.

SUMMARY

The ACM PMS is a research tool which will lead to improvements in the effectiveness of the IOS. The development of performance measurement for air combat training is expected to improve the instructor's capability to evaluate student performance. The development of effective visual displays of the performance measures will support the instructor in monitoring the student during the training event. Immediate feedback and instruction during the event would be expected to become more standardized and consistent. The display of the measurement information to the student and to the instructor during debrief will lead to improvements in the feedback that the student receives.

REFERENCES

1. Warner, H. (In Press). Instructor/Operator Station Handbook for Aircrew Training Devices. (AFHRL-TR-87-XX). Williams Air Force Base, AZ: Operations Training Division, Air Force Human Resources Laboratory.

2. Charles, J.P. (In Press). Instructor/Operator Station Design Guide (AFHRL-TR-87-XX). Williams Air Force Base, AZ: Operations Training Division, Air Force Human Resources Laboratory.

3. Polzella, D.J. (1983). Aircrew Training Devices: Utility and Utilization of Advanced Instructional Features (Phase I - Tactical Air Command). (AFHRLTR-83-22, AD-A135 052). Williams Air Force Base, AZ: Operations Training Division, Air Force Human Resources Laboratory.

4. Polzella, D.J. (1985). Aircrew Training Devices: Utility and Utilization of Advanced Instructional Features (Phase II - Air Training Command, Military Airlift Command, Strategic Air Command). (AFHRL-TR-85-48). Williams Air Force Base, AZ: Operations Training Division, Air Force Human Resources Laboratory.

5. Polzella, D.J. and Hubbard, D.C. (1986). Aircrew Training Devices: Utility and Utilization of Advanced Instructional Features (Phase III - Electronic Warfare Trainers). (AFHRL-TR-85-49). Williams Air Force Base, AZ: Operations Training Division, Air Force Human Resources Laboratory.

6. Polzella, D.J. and Hubbard, D.C. (In Press). Use of Advanced Instructional Features on Aircrew Training Devices: Summary Report. (AFHRL-TR-87XX). Williams Air Force Base, AZ: Operations Training Division, Air Force Human Resources Laboratory.

7. Easter, A.W., Kryway, J.T., Olson, W.R., Peters, S.M., Slemon, G.K., and Obermayer, R.W. (1985). Instructor Support Feature Guidelines. (AFHRL-TR-8557II). Williams Air Force Base, AZ: Operations Training Division, Air Force Human Resources Laboratory.

8. Pruitt, V.R., Moroney, W.F., and Lau, C. (1980) Energy Maneuverability Display for the Air Combat Maneuvering Range/Tactical Training System (ACMR/TACTS). Special Report 80-4, Naval Aerospace Medical Research Laboratory, Pensacola, FL.

9. Oberle, R.A. and Naron, S.E. (1978) The Air Combat Maneuvering Range Readiness Estimation System (ACMR RES). Project Overview. Center for Naval Analyses, CRC 355-Vol 1.

10. McGuinness, J., Bouwman, J.H., and Puig, J.A. Effectiveness Evaluation for Air Combat Training. (1982) Proceedings of the Inter-Service/Industry Training Equipment Conference.

ABOUT THE AUTHORS

Dr. Amanda M. Williams is a member of the Technical Staff with the Tactical and Training Systems Division of Logicon, Inc., where she has been involved in IOS design, training systems analyses and ACM research. She was formerly the Computer-Based Training Specialist for the F/A-18 Pilot Training Program. She has published in the areas of visual perception, applied visual perception, IOS design, performance measurement and computer-based training.

Dr. Gary S. Thomas has been employed in Army, Navy, Air Force and university research laboratories, and currently is a research psychologist at the Air Force Human Resources Laboratory at Williams Air Force Base, Arizona. He has responsibility for the design and development of instructor/operator stations, performance measurement systems and expert systems. He has 30 professional publications and paper presentations in the areas of learning, memory, training and instruction.

TRAINING ENGINEERING:
A PARAMETRIC APPROACH TO COMPUTER-BASED TRAINING DESIGN

Mary Stoddard Trainor and Andrew E. Andrews
Cognitive Engineering Design and Research Team (CEDAR)
Military Systems Group
Mail Stop F601, A-6
Los Alamos National Laboratory
Los Alamos, New Mexico 87545

ABSTRACT

Training engineering, a new model for computer-based training (CBT), has been devised and put into use by the Cognitive Engineering Design and Research Team (CEDAR) at Los Alamos National Laboratory. Training engineering is the application of scientific principles to the design, construction, and operation of efficient training systems. Such an approach is necessary because of the level of complexity CBT design and development has reached with the new advanced technologies. Instructional designers are under pressure to implement these new technologies more rapidly than has been required in the past, yet few models have emerged to aid designers in this process. Training engineering is such a model. It provides techniques for design and development that are derived from successful engineering techniques. This paper begins with a discussion of the engineering approach and then applies this approach to training. Examples from prototype CBT projects at Los Alamos are used throughout to illustrate the training engineering concept.

INTRODUCTION

This paper addresses the issue: How can CBT designers and developers keep up with new technologies and provide sound instructional approaches which meet user/project requirements?

This issue will be tackled using the approach called training engineering (see Fig. 1). The engineering approach includes selecting the right tool for the right job. The tools in this case include not only software and hardware but also different design and management approaches.

Over the past 20 years, our knowledge of effective designs for CBT has emerged to the point that we have learned where CBT works and where it does not. We have also learned that computers alone cannot solve existing training or performance problems and that extensive needs assessment and careful design are necessary if CBT programs are to be successfully implemented.

Although our knowledge has increased greatly in the area of CBT design, the hardware and software technologies supporting CBT have advanced at a much greater pace. Now we have relatively low-cost personal computers with the same computing

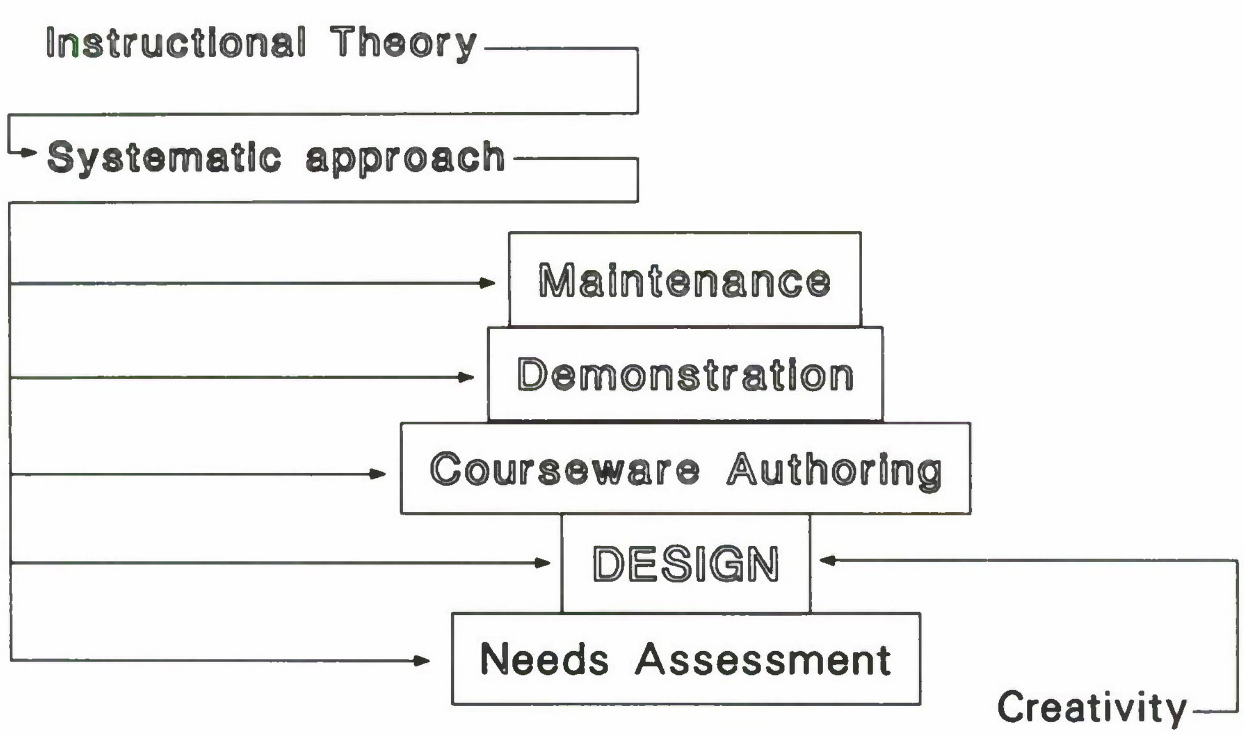


Fig. 1. Computer-Based Training Engineering

*This work was partially supported by the Army Research Institute.

power available only from large mainframes two decades ago. We have software tools and systems to facilitate the authoring process, enabling the development of much more sophisticated systems in less time and by less highly trained personnel. We have the capability of storing massive amounts of information which is very rapidly retrievable with a personal computer from storage media such as CO ROM. We can see and hear realistic problem solving scenarios and then work with them in an educational context via interactive videodisc and digital audio. High quality color graphics and animation are able to provide the fidelity of actual video when video is not obtainable or cost effective.[1]

Since the new hardware and software technologies have been so well publicized in the academic and popular press, managers in government and industry are increasingly demanding use of the new technologies. They read the ambitious promises of the new announcements and become convinced that all their training and performance problems can be solved if they only purchase products X, Y, and Z.

Rapid implementation of these hardware and software advancements by designers has resulted in a need to make design and development approaches used for CBT more responsive to the complexity of the task. Two examples of areas requiring new approaches are hardware and software selection and screen design. The selection of hardware and software for a particular development project is now much more complex than in the past, requiring different skills on the part of the staff and often a greater time commitment. For example, there are currently over 400 authoring systems from which to choose.

The challenges in designing an easy-to-learn and easy-to-use system are changing drastically because of the complexity of both function and choice added by the new technologies. Guidelines we have relied upon in the past for screen design are becoming obsolete[2,3] and are not able to be replaced before a new, more powerful technology emerges. Training engineering should provide a beginning for the evolution of new approaches, which are more responsive to this newfound complexity. This complexity is here to stay, as is CBT; and, therefore, investment into a training engineering outlook should be cost effective from a management standpoint.

Training engineering is both a procedural approach and a philosophy. On the philosophical level, training engineering embraces practicality, pragmatic decision making, building or creating new things, design, utility, and a goal orientation. It includes an attitude that approximation is acceptable initially and precision is achieved through iteration.

This paper begins with a general discussion of the engineering discipline. This is followed by a description of a structural engineer's approach to a construction problem and a training engineer's approach to a CBT problem. The training engineering approach is then summarized.

ABOUT ENGINEERING

The discipline of engineering is a proven one. Some of the greatest human accomplishments have been built by engineers: the Taj-Mahal, the Golden Gate Bridge, the Hoover Dam. These accomplishments reflect the building of extremely complex systems, which needed to be built to last, be on schedule, be built to specifications, and be built to be aesthetically pleasing. CBT today is becoming almost as complex! How were such complex engineering projects accomplished and what can CBT learn from them?

Engineering requires a systematic approach to design, construction, and project management. In the field of engineering, design is based upon proven theories from which the systematic approach is derived. Because of the large number of variables to choose from and the large number of decision points, a systems approach is necessary. Nevertheless, the engineer who exclusively follows a systems approach does not usually succeed. The product of his engineering skills may be structurally sound, but it may be aesthetically displeasing to the human eye.

The systems approach is not new to the field of instruction,[4] but the theoretic basis for CBT was slim until fairly recently.[5] The number of studies that have been performed in the area of computers in instruction is now massive, compared to even a decade ago. Therefore, the basis for the systematic approach is now more sound, enabling us to take the bold step towards an engineering of training. Although CBT design itself is not yet a science, it does have an evolving methodological base from which one can work systematically. The field of architecture is not a science either, yet it is a vital component to engineering projects. Early CBT was either direct conversion of an existing course onto the computer or an art form. Now CBT can be, like engineering, based on a solid foundation and yet also leave room for creativity.

In comparing training engineering to the structural engineering field, one can see the following analogies in terms of roles:

Architect + Structural Engineer + References on Materials = Instructional Designer + Software Developer + Subject Expert

The instructional designer's role, however, encompasses tasks performed by the architect as well as the structural engineer. This role is illustrated more clearly in the section following.

The training engineering approach also puts the training department in a mode of being requirements-driven; it helps avoid the pitfall of choosing to do CBT just to do CBT! An engineer would not choose to build a bridge across a river just because it was an attractive location for a bridge; he/she would require a well-demonstrated need as well as adequate funding. In addition, the bridge would not be built with a pedestrian path and six lanes if it was in a low population area. Yet today, many training departments are implementing CBT on inappropriate applications with hardware and software configurations, which do not match the user needs.

A STRUCTURAL ENGINEER VS. A TRAINING ENGINEER

To define training engineering more clearly, it is useful to go through a scenario of a structural engineering example and then follow it with a scenario of a training engineering example. Table I summarizes the characteristics of each discipline.

TABLE I

STRUCTURAL VS. TRAINING ENGINEERING

	Structural Engineering	Training Engineering
Step 0	ANALYSIS	NEEDS ASSESSMENT
	o sponsor, user interviews	o faculty, student interviews
	o observations	o classroom observations
	o attitude surveys	o attitude surveys
	o cost analyses	o cost, facility analysis
	o forecasting of future needs	o desired future system
	o scheduling requirements	o scheduling requirements
Step 1	DESIGN	INSTRUCTIONAL SYSTEM DESIGN
	o knowledge of materials	o software/hardware info.
	o site, geology information	o info. on possible applications
	o site selection	o application selection
	o analysis of site	o knowledge base study
	o cost estimate	o cost, resource estimate
	o structural design theory	o instructional design theory
	o approaches to bridge building	o instructional strategies
	o aesthetics	o creative designs
	o proposed time schedule	o proposed time schedule
	o design plan draft	o design document draft
	o review and revisions	o review and revisions
	o model building, sketches added	o rapid prototyping
	o review and revisions	o testing and revisions
	o final design plan	o final design document
	o approval	o approval
Step 2	CONSTRUCTION	COURSEWARE AUTHORING
	o manage project	o manage project
	o purchase materials	o purchase hardware
	o hire workers	o add to staff as required
	o assemble team	o divide labor
	o create quality assurance plan	o evaluate plan
	o build bridge	o write courseware
	o provide reports to sponsors	o report to sponsors
	o adjust schedule as needed	o adjust schedule as needed
Step 3	GRAND OPENING	DEMONSTRATION
	o clean up for opening	o debug and test
	o coordinate ceremony	o coordinate briefing
	o write script, review	o write briefing, review
Step 4	MAINTENANCE	MAINTENANCE
	o safety check	o ongoing evaluation
	o maintenance plan	o maintenance plan

Structural Engineer's Step 0: Analysis

A small town in North Dakota had a bridge which several hundred people travel across to get to work each day. This bridge, built in 1925, was wooden and was judged as structurally unsound. It was critical that the bridge be available at all times for the economy of the town. It was a two-lane bridge, but it did not adequately handle the traffic flow during the morning and evening rush hours. In addition, because of the growth of the town during the past 60 years, pedestrians needed to travel across the bridge to shopping centers and residential areas. The town council ordered and paid for an analytic study to be performed to determine the following:

- The amount of money the town could realistically afford for the the new bridge,
- The requirements of the bridge load today and 10 years from now,
- Resident's attitudes about the new bridge location,
- Town council and chamber of commerce attitudes about desired specification, and

- The desired time scale, to minimize inconvenience.

This information was complied, recommendations were made, and a report was submitted to the town council for review.

Training Engineer's Step 0: Needs Assessment

A military college determined that it needed to integrate the use of computers in its curriculum. A general, who saw the emerging role of computers in every facet of the armed forces, was concerned about the college's not adequately preparing officers to use computers in the battlefield of the future. Therefore, several members of the faculty started attending short CBT courses, conferences, and expositions to learn more. As more of the faculty members gained expertise in the area of military science, they soon recognized that they needed to consult with some external experts in the area of CBT before they committed significant resources. Consequently, they hired an independent institution to perform a needs assessment.

In the needs assessment, the following was done:

- The education/training goals of the college were identified.
- The current training system was characterized, through observation of classes and interviews with faculty and students.
- Desirable features for the optimal training system were identified through attitude surveys.
- The current and desirable future systems were compared, and the differences in project requirements were described.[6]
- The cost and facility constraints were studied.
- The schedule was reviewed.
- A needs assessment report was prepared and submitted for review and use by the college.

Structural Engineer's Step 1: Design

The town council reviewed the analysis report and accepted its recommendations. Its first recommendation was to go out on bid for an engineering firm to design the bridge and coordinate its building. These steps were done and the engineers were on board. The head engineer, Joe Fraser, was responsible for producing a design plan for the project. For this, he relied upon his own knowledge of structural engineering, as well as knowledge he had gained from the analytical study and other sources. Specifically, he used information on:

- Characteristics of different building materials
- Current construction costs
- Physical features of different possible sites for the new bridge
- Geological characteristics of the area
- Quality control approaches
- How to build a bridge across this type of river with this type of span and required load
- Maintenance alternatives
- Aesthetically pleasing vs. displeasing bridge designs
- Required time scale for such a project
- Contents of a design plan for a bridge.

Mr. Fraser worked with an architect to pull together the specifications and complete the preliminary design plan. This plan included models and artist's sketches of the bridge. Because Mr. Fraser was a licensed engineer, he had to make certain that his plan reflected his skills and instilled trust in the reader. His plan was reviewed by the engineering firm internally and then revised. It was then taken to the town council for preliminary review, comments were collected, and it was again revised. The plan was then made public to the town, and comments were received at a town meeting. Final revisions were then made.

Training Engineer's Step 1: Instructional System Design

The needs assessment study was reviewed and accepted by the college, and the first recommendation (to assemble a design and development team) was implemented. This team consisted of three people initially, an instructional designer, a hardware expert/software developer, and a subject matter expert (rotational duty). This team was responsible for the first phase of the project, development and testing of one CBT course to replace an existing course for which there was an instructor shortage and a stable subject matter.

The instructional designer, Sara Long, set out to write a design document. For this design document, she relied not only upon her own knowledge of CBT design but also studied the recent literature for new approaches which might suit the needs of this project. She consulted with the hardware/software expert on the optimal configuration to use here and sent this expert out to various expositions to critique current technologies and report back to her. She consulted with the subject matter expert and they systematically selected an application, which would have a high early payoff. The subject matter expert researched the knowledge base for the chosen application and discovered widespread discontent with the current course curriculum; students claimed little of the classroom training was transferable to the field. In her cost/resource estimate and schedule, she factored in an analysis of the knowledge base into the design time. She planned to examine the conceptual model upon which the current instruction was based.

In addition, she examined user requirements for creative interactivity methods, user interfaces and instructional strategies to use in the designing. An instructional strategy is the pedagogical method used in a CBT lesson to aid the student in mastering the performance objective. There are many taxonomies used for instructional strategies. The one she used is by Alessi and Trollip,[7] in which five different instructional strategies are identified: tutorial, drills, simulations, instructional games, and testing. The instructional strategy chosen is dependent upon the expected outcome. For example, if new knowledge must be acquired by the student, then the tutorial strategy is often chosen. She chose a simulation instructional strategy. Because the need for positive transfer of training was so great for this application, students had the requisite fundamental knowledge in the area and the skills required lend themselves to scenario-driven exercises. The key here, she knew, was to select an instructional strategy which matched user requirements and test it before the actual development began.

Ms. Long then compiled the design document, which contained a preliminary CBT lesson design and various flow diagrams. The design also was built with a separate knowledge base from the user interface, facilitating maintenance. Although there is no licensing of instructional designers, Ms. Long was bound by a moral commitment to produce a design based upon sound instructional principles. Therefore, she made sure that the design document reflected her skills and ethical principles. Following completion of the document, she had the college review the design document and provide feedback. During the review she and the subject matter expert performed an analysis of the knowledge base and incorporated this information along with the reviewer's changes in the next iteration of the design document. She then worked with the hardware/software expert to bring up a prototype very rapidly (two months) on borrowed hardware. Rapid prototyping facilitates the highlighting of desirable features and pinpointing of potential problem areas at an early stage in the project. Research has shown that systems which have been rapidly prototyped result in much lower maintenance costs.[8]

The initial prototype was then tested by the project team, revised, and demonstrated to a few highly interested faculty members. This group included one of the former instructors of the course. These faculty members reviewed the prototype and provided not only subject matter comments but also comments on the user interface, the methods of interactivity chosen, the hardware configuration and the instructional strategy. The simulation instructional strategy received rave reviews, even from the former instructor, as did the use of the borrowed videodisc for scenario presentation. Only a very few parts of the borrowed videodisc were used for the prototype, and a new videodisc needed to be made if this technology was selected. The cost of the design of a simulation, as well as the videodisc, was outlined in the design document. Faculty feedback was discussed by the project team. The hardware used for the prototype was reviewed, and the configuration recommended was revised, deleting the use of digital and audio because of less expensive storage on the videodisc. The borrowed hardware was returned. The design document was revised and submitted for approval.

Structural Engineer's Step 2: Construction

When the design plan was approved by the town council, the engineering firm geared up to begin actual construction. At this point, the design was frozen. They mapped out the project systematically, defining the separate phases and then determining when the phases need to converge on the schedule. The phasing highly affected the workers on the project and the materials at the site at any one time. A project management system was used in determining the schedule, with the major and minor tasks and their milestones clearly identified. The schedule reflected the separate phases and which ones could occur concurrently.

Following the mapping out of the schedule and tasks, the following were performed:

- Materials were purchased.
- A staggered schedule for materials delivery was arranged for to minimize any loss from burglary at the site.
- Workers were interviewed and hired, to match needs at the various times of the project.
- Teams of workers were assembled, with the team foreman oriented concerning the supervisory approach taken by this particular firm.
- The quality control plan was enforced, to ensure that following the project completion the amount of maintenance required was minimal and the degree of safety was maximal.

Actual construction was then performed with Joe Fraser managing the various foremen and ensuring that the project management chart was kept up to date. He kept the town council informed regarding progress relative to the announced schedule. When resource estimates were found to be inaccurate, he reported to the town council and sought advice prior to implementing the change. He performed quality checks following completion of each task.

Training Engineer's Step 2: Courseware Authoring and Production

The design document was approved, and hardware was procured for development and testing. The design was reviewed once again, with the understanding that any changes from here on would probably adversely affect the schedule. Sara Long then laid out the schedule, divided the required labor among the staff, and set milestones for interim demonstrations, testing, in-progress reports, and documentation.

Incorporated in this schedule was work being done in parallel, in particular the programming work being done at the same time as the videodisc production. Since her staff did not currently include script writing and video production expertise, she sought part-time employees for these tasks. When these employees were on board, she had weekly team meetings to keep informed of their work and to orient them concerning her management style. These meetings primarily ensured coordination and creative interaction among the team members and also enabled Sara to write monthly progress reports to the college's upper management.

As Sara had experience in educational evaluation, she also wrote up an evaluation plan. This plan included not only formative evaluation, internal testing and pilot testing during the project but also a summative evaluation plan for the college at the conclusion of the project. The formative evaluation ensured that the software was responsive to the needs of users, as well as being bug free. The summative evaluation was used for decision making. Because the software developer concentrated on development and not testing, she made herself responsible for ensuring that the evaluation plan was followed.

When the building of the courseware began, issues arose on a almost daily basis, which required reviewing the estimated schedule and resources, as well as the basic design. When changes were determined to be in the best interest of the final product, Ms. Long presented those to her management prior to implementation.

Structural Engineer's Step 3: Grand Opening

Once the construction was completed, the bridge was ready to be dedicated prior to use by vehicular and pedestrian traffic. This was the big day that really made the project worth all of the hard work, giving each team member a feeling of personal accomplishment. The engineer coordinated

the grand opening with the town council and supervised the workers in performing last minute clean up. Mr. Fraser himself was asked to give the short dedication speech, along with the mayor, prior to the dedication. Therefore, he wrote a script and had it reviewed by his firm, as well as the mayor. This project was visible statewide, as the media had targeted it for attention; and, thus, the engineering firm should attract new business if all went well.

Training Engineer's Step 3: Oemonstration

Once the courseware was built and tested internally and with a few target users, it was ready for full-scale demonstration to the college faculty. This project was the first CBT project, setting the tone for the rest of the CBT development effort for the college. Therefore, this demonstration was critical. It was also critical to the job security of the project team. The courseware had to work; the briefing associated with the courseware had to be polished. The entire team was involved in preparations: the instructional designer with writing the briefing and scripting a demonstration that revealed the strongest selling points of the courseware, the hardware/software expert with debugging and preparing the demonstration, and the subject matter expert with assuring that the content of the briefing was appropriate to the target audience and with setting up the briefing.

Structural Engineer's Step 4: Maintenance

An area no one wanted to think about because of its lack of glamour is: Who worries about the bridge after it is done? In fact, a small town like this one did have some civil engineers who did day-to-day maintenance on town buildings and properties, but they had little experience with inspecting and performing upkeep on a bridge. Therefore, as part of the project, a maintenance plan needed to be drawn up. This plan needed to include not only specifications on who performs safety checks and how often but also on how both major and minor repairs were to be handled. This plan needed to be compiled with information from the town council, specifying their preferences. Maintenance was first considered in the design phase, and therefore this step was a matter of implementing a plan tentatively made earlier.

Training Engineer's Step 4: Maintenance

The maintenance of courseware is becoming a hot issue in the CBT field, and the instructional designer was, therefore, aware of the need to provide a maintenance plan along with the final project report. The demonstration was successful, and she would be able to keep her position along with the rest of the team. They would be available to perform maintenance, but they have moved on to another development project and will not have time allocated for maintenance. Consequently, her maintenance plan required the college to devote a subject matter expert, who is computer literate but not a programmer, to perform the maintenance. Because of anticipation of maintenance demands at the design stage, the courseware was built to facilitate rapid updating of the knowledge base. This kind of maintenance is the most common type, and the subject matter expert is very capable of performing the task. The maintenance plan also included recommendations for software revision or debugging. In this case, the original design team would be called upon.

SUMMARY

In reviewing Table I, one can quickly observe how similar structural and training engineering are in practice. Although the title of the major steps may differ somewhat, the subtasks are very similar in function throughout. This fact highlights the use of applying an engineering approach to the development of computer-based training.

In addition to comparing training and structural engineering, one can step back and examine it in light of other approaches. The key to such a comparison is defining the scope of training engineering. Training engineering is not a new instructional design model or a new project management methodology, but rather it is a comprehensive, high-level model for integration of new technologies. It is responsive to the new, added variables with which the instructional designer now has to deal.

Nevertheless, training engineering is also a philosophy. It is a pro-active approach to problem solving that requires working within acceptable risks while striving for creativity. It means using available tools in new ways and recognizes that while one strives for perfection, it is not required for success. It stimulates an urgency to produce tangible products that can be examined and revised. And, finally, it clearly recognizes the responsibility of the engineer to produce sound and enduring products. Table II summarizes these points as five principles to be followed.

TABLE II

FIVE PRINCIPLES FOR IMPLEMENTING TRAINING ENGINEERING

- Be sensitive to design-aesthetics; they are important!
- Use tools and materials available today, not the promise of what is in the laboratory.
- Accept an approximate solution, within safety tolerances, as a good solution.
- Prototype and iterate.
- Remember your responsibility for sound construction.

This paper provides only a beginning to the development of the training engineering concepts, or elaboration of the engineering approach to the various subtasks within needs assessment, design, and authoring is necessary. It does, however, reorient a designer's thinking and enables us to leap to a new level of accomplishment. Instead of researchers' having to develop a meta-level approach to accommodate the new technologies, a proven discipline that maps over well to the education and training field is relied upon to help us make that leap.

REFERENCES

1. Andrews, A. and M. Trainor, "Constructing a Computer-Tutor," Data Training, 28-34, February, 1987.

2. Heines, J., Screen Design Strategies for Computer-Assisted Instruction (Digital Press, Bedford, MA, 1984).

3. Smith, S. and Aucella, A., "Design Guidelines for the User Interface Computer-Based Information Systems," Mitre Corporation technical report 8857 for the Air Force Systems Command, ESD-TR-83-122 (April, 1983).

4. Davis, R., Alexander, L., and Yelon, S., Learning System Design: An Approach to the Improvement of Instruction (McGraw-Hill Book Co., New York).

5. Reigeluth, C., Instructional-Design Theories and Models: An Overview of their Current Status (Lawrence Erlbaum Assoc., New Jersey, 1983).

6. Rosett, A., "Front End Analysis," handout from a presentation at the Training Conference, San Francisco (1985).

7. Alessi, S. and Trollip, S., Computer-Based Instruction: Methods and Development (Prentice Hall, Inc., New Jersey, 1985).

8. Connell, J. and Brice, L., "Rapid Prototyping," Datamation, August 15, 1984.

ABOUT THE AUTHORS

MARY STODDARD TRAINOR is a technical staff member of the Cognitive Engineering Design and Research Team (CEDAR), Military Systems Group, Los Alamos National Laboratory. She provides expertise in instructional design, user interface design, computer-based training, and cognitive science for CEDAR. She has been a principal contributor to the development of the computer tutor concept. Trainor has been at Los Alamos for six years, with prior positions in the computer user education, documentation, and information systems areas. She holds a Ph.D. from the Group in Science and Mathematics Education at the University of California at Berkeley. Trainor has 25 publications, with her most recent in the areas of intelligent tutoring systems, computer tutors, and human factors in computing.

ANDREW E. ANDREWS heads CEDAR Team, Military Systems Group, Los Alamos National Laboratory. The CEDAR Team has pioneered the concept of the Computer Tutor, a bridge between the computer-based training and the artificial intelligence communities. Andrews holds an M.S. in Nuclear Engineering from the University of California at Berkeley, an M.B.A. from Long Island University, and a Master of Military Arts and Sciences degree. Prior to his work in CEDAR, he held varied positions in the U.S. Army that included exercise and training development. Additionally, he served as a military research associate at the Los Alamos National Laboratory and culminated his Army career as the senior air defense staff officer at the Supreme Headquarters Allied Powers Europe. Andrews has several publications in engineering, military doctrine, and training technology.

EFFICIENT, PRODUCTION-ORIENTED CBT AUTHORING

Kevin J. Parks
Ford Aerospace & Communications Corporation
Training Systems Center
10800 Parkridge Boulevard
Reston, Virginia 22091

ABSTRACT

Computer based training is no longer an experimental method of training in the military. It has been used on a very large number of programs, either as initial training to precede simulator training or as standalone training. There has, however, been much controversy over how best to produce computer based courseware. The training community has realized that a major cost in the use of CBT is the development of the courseware. The goal is to develop the most effective courseware for the least cost. The controversy has been between whether to use an authoring system, which speed production and is easy to use but has restrictions, or to use an authoring language, which is more difficult to use but provides more capability and flexibility.

This presentation will describe a solution to that controversy, the use of an authoring package that combines both an authoring system and an authoring language. The package was designed to be multilevel so that ease and power would both be available to the courseware developer.

The first level of the authoring package is designed to be easy to learn and quick to use. It Is intended for the beginning author and the development of simple courseware interactions. It consists of a series of menus and forms that the author uses to specify how the courseware will work when the student interacts with it.

The next level is designed to be used when the author needs more sophisticated tools than are available in the first level. The difference is that there are more menus and more choices.

The third level is an authoring language that provides extremely powerful tools for developing sophisticated part-task training and simulations.

INTRODUCTION

Computer Based Training (CBT) has been accepted as a major component of the entire training package in military training. For example, CBT is used to train maintenance crews for the M-1 Abrams tank, flight crews for the F-18 and the S-3, language experts at the Defense Language Institute, and undergraduate pilots in the Air Force, to name just a few applications.

The issue is no longer to use or not to use CBT. The issue is how to produce excellent courseware in a cost effective manner. The military and others have become aware during the last several years that the real costs in CBT are in the courseware development, not in the computer hardware. There are two basic schools of thought about how to produce low-oost courseware: use an authoring language or use an authoring system. To understand the issues, we must examine two factors that affect costs. First, we need to look at developmental efficiency. Second we need to look at the characteristics of authoring systems and authoring languages and how those characteristics affect developmental efficiency. After examining these two factors, we will look at one solution to cost-effective courseware, a three-level authoring system which also includes, as an intergral foundation, an authoring language.

DEVELOPMENTAL EFFICIENCY

The development of CBT courseware involves many people and many different tasks. Coordinating the development tasks, matching people and tasks, is not a small job. How well it is accomplished often determines how efficiently the courseware material will be developed. What you choose as your development tool (authoring system or authoring language) plays a large part in the job of matching people and skills.

Staff Skills

Let's look first at staff skills. In any large-scale CBT development project, the staff must have a range of interests, talents and skills. A properly balanced staff will include:

- o subject matter experts
- o instructional designers
- o instructional programmers or developers
- o graphics programmers or illustrators.

Generally, subject matters experts and instructional designers are high cost people who must be on the project, they must be used efficiently and effectively to save money.

Instructional programmers are also high cost; however, if they can be replaced by instructional developers with minimal programming experience, large savings may be realized.

Illustrators are also lower cost staff than graphics programmers and can be used to reduce costs. What is more, illustrators produce better graphics than programmers and these make the instruction better. But in order to use illustrators, the system must permit graphics to be produced without programming.

To sum up, in a large-scale CBT development project, we must use high-cost subject matter experts and designers and we must use them efficiently. This means we should use them only to do their particular jobs of designing and producing the instruction, not as on-line programmers. We do not, however, have to use high-level, high-cost programmers. If we can substitute developers and graphics illustrators for the on-line development effort, we can save considerable costs.

Courseware Development

Just as the staff is not homogeneous, the courseware to be developed is not all the same either. Courseware ranges in complexity from simple cognitive learning to complex simulations and part task training exercises. Any body of courseware encompasses all ranges of the spectrum.

The cost of development ranges from low to high also. As a general rule of thumb, it would seem that simple courseware would be quick and easy (cheap) to produce and complex courseware would be more difficult and take longer (expensive); however, this is not always the case. A lot depends upon the tool you are using to develop the courseware.

If, for example, you are using an authoring system that only allows you to develop four-item

multiple-choice questions and you must have a fill-in-the-blank question, you might have to do extensive programming to get around the system's restrictions. Or you might want to allow misspellings, but the system has no spelling algorithm and you have to put in all the acceptable misspellings as correct choices. Or you may have to put in ten displays to explain a complicated concept, but, the system only allows eight displays and you must break up the lesson into multiple parts to get the number you need.

Using an authoring system with limited capabilities and many restrictions may not permit you to produce the courseware you require.

On the other hand, if you are using an authoring language with a lot of power and flexibility, you may be forced to use complex programming to do very simple tasks such as judge a multiple-choice answer as right or wrong. Or you may need many lines of code to give different feedback messages for each multiple-choice distractor. Using an authoring language may cost you too much when producing the simple parts of the courseware.

Cost efficiency depends heavily on the tools used to produce the courseware, as well as on the staff doing the production. Both factors affect developmental efficiency, and thus costs. The tools we choose for the on-line development effort play a major factor in our ability to produce CBT courseware in a cost effective manner.

AUTHORING LANGUAGES

Now, let us look at the characteristics of authoring languages and authoring systems and see how they affect costs.

An authoring language is a type of programming language and as such has many of the same characteristics. It may, in fact, be a programming language that has been enhanced to specifically support CBT development or it may be a language specifically designed for the production of courseware.

Authoring languages range from the relatively easy to use to very difficult. They provide basic capabilities. How well these capabilities are used depends upon the developer's or programmer's skills.

Authoring languages are sophisticated and have many capabilities. The developer can do almost anything he wants. However, he may need specialized training and a lengthy "apprentice' period. The real statement is "The developer can do what he wants as soon as he figures out how to do it." The cost/efficiency balance is between the cost of a novice, who will take time to learn to use the language, and the cost of an experienced programmer, who may be able to develop instruction quickly. Authoring languages are best used by people who have some programming skills.

Authoring languages are flexible. There are often several ways of doing the same thing. This sounds good, except when you stop to think about control of the authoring process, i.e., coordinating and intergrating the product of multiple developers.

Authoring languages have advantages in sophistication, capability and flexibility. These may also be disadvantages in the cost-efficient production of courseware. There is such a thing as too many bells and whistles.

AUTHORING SYSTEMS

Now let's look at the characteristics of authoring systems as they compare to authoring languages.

Authoring Systems are software packages that are specifically designed for developing CBT material. They also range from the very restricted to the very capable. They are generally advertised as "Easy to Use, No Programming Needed, Low Cost, etc." The courseware developer must ask at what cost are they easy to use.

The hidden cost you pay is in restricted capabilities that authoring languages have. However they make up for this in taking care of many of the programming details for the developer. The fewer details you have to take care of, the more efficiently you can author. But, you have to be able to live with the restrictions; therefore you want the restrictions to be as few as possible.

A simple-to-use authoring system may come with many restrictions.

- o You may have only a certain area on the screen for text and a limited number of display pages available.
- o Graphics may be limited to line drawings or certain resolutions and may or may not have colors.
- o Video and the ability to overlay computer generated materials on the video may be restricted.
- o Branching to different parts of lessons may be limited. Multiple branches may not be possible.
- o You may be restricted to certain types of questions such as multiple choice and true/false questions.
- o Feedback messages may be limited to correct or wrong and may not be answer specific.
- o No course structure may be available. There may be no way to link lessons together.
- o No instructional templates may be available. There may not even be a way to use templates.
- o There may be no way to communicate with peripheral devices through the system.

The real problem is what happens when you want to do something that the system does not support. The developer has several choices:

- o change the design
- o use a different authoring system
- o use an authoring language
- o program what must be done

This is not an easy decision to make in the middle of a courseware development project, particularly if you want to keep costs down.

Using an authoring system allows you to use developers instead of programmers to do the on-line production of material, it allows you to produce material sooner; but, it restricts your instructional capabilities.

A THREE-LEVEL AUTHORING SYSTEM

What, then, is the solution to this problem of authoring systems, authoring languages, and low-cost courseware development? One solution has been developed by Ford Aerospace and Communications Corporation. We are going to look at how ADAPT, a multi-level authoring package, can aid in reducing the costs of courseware production.

Efficient production of courseware can be defined as producing the most effective courseware for the least cost. To do this, you have to deal with a staff that has varying authoring capabilities and courseware design of varying levels of

sophistication and complexity. Let us look at how a multi-level authoring package will help do this.

The first level of the authoring package is the beginning level. It is designed for the novice developer. It is menu driven, i.e., what you see listed on the menu is what you can do. This authoring level has two basic goals. First, it teaches the beginning developer how to use the authoring system very quickly. In a few hours, he can produce real courseware. Second, it allows any developer to produce simple, but usable courseware very rapidly.

The fact that Level 1 is designed for the novice developer and is easy to use does not mean that he cannot produce instructionally sophisticated material with it. An example of this is video support. The basic requirements for using video are

- specifying what frames to play,
- specifying what audio channel to use, and
- specifying what speed to play.

These can all be specified using a simple menu; therefore, Level 1 supports the use of video for instruction.

Being simple to use does not mean that you cannot have complex interactions. If you can break the instruction down into discrete steps and the number of steps does not exceed the capability (capacity) of the level, the instruction can be authored using this level.

For example: Suppose a panel simulation has ten different steps. The developer has to construct ten individual pages, one for each step of the panel.

The second level of the authoring system builds upon the first level. It is also menu driven. It uses the same editor structure and many of the same menus. The only difference is that there are more menus, and they have more choices. In this level, many of the restrictions which were present in the first level are lifted. The developer has increased capabilities and can author more complex instructional material.

For example, in Level 1, all questions are worth one point. In Level 2, the developer can assign as many points to a question as he wishes.

The panel simulation (finite state table) which took ten pages in Level 1 can be done with one page in Level 2.

Text can be overlayed upon existing text as the student moves through the courseware material. In Level 2, the developer can take greater control over the CBT system if he so desires.

The third level is an authoring language. This level also builds upon the previous levels. The editor structure is maintained and, where appropriate the same menus are used. At Level 3, the developer has all the capability and flexibility of the authoring language. He can do anything he wants (that can be done by the computer) as soon as he can figure out how to do it.

In addition to the same editor structure and menu carry-over, the three levels have additional relationships. Levels 1 and 2 are menu driven. Levels 1 and 2 are also code generators. As the author fills out the menus, the data are used to construct level 3 authoring language code. The CBT system only sees Level 3 code. The CBT system does not care at what Level (1,2, or 3) the courseware was authored. To the system (and the student), there are no differences among Level 1, Level 2, or Level 3 courseware. This has many implications for the efficient production of courseware.

THE THREE LEVEL AUTHORING SYSTEM AND EFFICIENT COURSEWARE PRODUCTION

One of the factors affecting efficient courseware production is the length of the learning curve for the novice developers, that is, the length of time it takes before they can produce usable courseware material.

When using an authoring system, there is generally a short learning curve before you reach production level skills. The drawback is that you may soon reach the upper boundary of the system capabilities. When using an authoring language, the learning curve is longer before you attain production level skills. You may never be able to reach the upper boundaries of the system's capabilities, however.

With the three level system, there is a short learning curve for Level 1, the developer can reach level skills very quickly. When the feature he needs for the courseware exceed the upper boundary of Level 1 capabilities, he can easily move into level 2 thereby increasing the features of the materials. He can move into Level 2 with little or no additional formal training because he is already familiar with the structure of the system and its menus. As he continues to gain experience, he can move into Level 3 (the authoring language) and access all of the capabilities it provides. Moving into Level 3 means that he must learn the syntax and format for the authoring language. But, he does not have to learn a new editing structure and many of the menus from the preceding levels are still used.

The three levels in the authoring system are progressively related, (Level 2 builds upon Level 1 and Level 3 builds upon Level 2). When a novice developer has to do a more complicated lesson, he can easily move up a level in the authoring system. He already knows the structure and the common menus. He only has to learn the new menus and their functions, and he is ready to handle more advanced material. The structure of the system allows the developer to move up levels at any point in the development and to combine all levels within a single lesson or a whole course.

The three level authoring system gives courseware development managers a way to match authoring skills with development requirements. The courseware development manager can match the sophistication level of the courseware design and the experience of the developer when making work assignments.

Novice Developer	Simple Design	Level 1
Experienced Developer	Moderate Design	Level 2
Expert Developer	Complicated Design	Level 3

It is also possible, with the three level authoring system, to have multiple individuals working on a simple instructional design. The novice developer, using Level 1, can put in those sections which he can handle easily. The more advanced developer can then use Level 2 to enhance that material. At the same time, the expert developer (using Level 3) can put in the sophisticated simulations, which only he can do.

The three level structure also provides authoring task efficiencies. One of the time consuming tasks when developing courseware using an authoring language is debugging the code. One of the efficiency factors in a menu driven authoring system is that the system produces error free code. Levels 1 and 2 of our three level system are menu driven code generators. The developer fills in the menu and the authoring system then produces the level 3 code. This code is error free and requires much less testing during the debug phase.

What is more, the system automatically documents the code, by inserting remarks, so that it is clear to the developer what is occurring in the level 3 code.

The development of courseware is often an interative process. A prototype is designed, tested, and revised. This may go on through several process cycles. The three level authoring system supports this type of development process. Prototypes can be developed rapidly using Level 1. Because Level 1 is fast and easy to use. The initial development is not an expensive proposition. When the prototype has been tested and the developer is ready to produce final versions of the material, he does not have to start from scratch. He can enhance the Level 1 version with higher level material or he can upgrade all of the material to a higher level.

SUMMARY

A three-level authoring system is an effective tool for the cost efficient production of CBT courseware. It combines the best qualities of authoring systems and authoring languages while avoiding their individual problems.

The three-level authoring system allows individuals with varying skill levels to work on the same courseware. Project managers can match authoring skill to courseware development requirements.

The three-level system provides training support for itself. As an author increases his capabilities and moves up through the levels, he does not have to learn a completely new structure and format for each new level. Each of the levels builds upon the previous level.

The three-level system supports the prototype, test, and revise method of development. It is very easy to produce a prototype instructional section using Level 1. After testing, you don't have to start over during the revision stage; you can enhance or upgrade the prototype material.

The three-level system, called ADAPT, has now been in use in the field for several months and is showing its capabilities in the courseware development process.

ABOUT THE AUTHOR

Mr. Parks is an Instructional Designer at the Training Systems Center, Ford Aerospace and Communications Corporation in Reston, Virginia. He holds an MS in Education (Instructional Research and Development). For the last three years he has been involved with the design and development of ADAPT.

INTERACTIVE VIDEO: A PROJECT REVIEW WITH IMPLICATIONS FOR TRAINING IN THE BRITISH ARMY

Author: Major Tony Croucher MSc RAEC
Commanding Officer: Colonel Donald H Oxley MA
Army School of Training Support
Royal Army Educational Corps Centre
Wilton Park, Beaconsfield
Bucks, HP9 2RP, UK

ABSTRACT

The concept of Interactive Video (IV) is examined in the light of the training requirements of the British Army. The reasons for the IV project are detailed, together with the basis for the selection of the system, project implementation, subject identification and the courseware design processes. Difficulties in project management and in interactive design are discussed and a structured approach to the design process presented. This approach was based on the combined use of structured design methods, flow charts, and screen layout documents. The results indicated that the approach was valid, that effective interactive design was difficult, and team stability vital. The knowledge gained from the study suggests that in view of the extent of initial and continuing resource overheads, the military use of IV is likely to focus on such applications as simulation where cost benefits may be more easily identified.

INTRODUCTION

There are an increasing number of computer controlled video systems, commonly referred to as Interactive Video, now available for use as training devices. These vary in their capabilities and many are pro-moted as providing some form of student management, rapid access to high quality video pictures and trainee interaction. These systems have attracted considerable interest in both military and civilian organisations within the UK and are seen as being potentially powerful and effective training tools.

BACKGROUND

The training organisation within the British Army is constantly facing increasing demands upon the resources available to meet essential training needs. In 1982 the Army School of Training Support (ASTS) was tasked to review the use of Interactive Video in both military and civilian contexts, (1) and to investigate the military potential of low-cost tape systems based upon existing Army equipment. Tasking was extended in November 1985 to embrace an advanced disk-based IV system in order to assess the implications and potential of this new technology for use in Army Training.

There were a number of possible options considered in arriving at this extension of tasking. These included monitoring and/or involvement with suitable civilian and military projects sponsored by various Government departments. All of these options were rejected because they either did not reflect the needs of the Army, or did not exist.

SYSTEM SELECTION

Criteria

To meet Ministry of Defence (MOD) criteria and guidelines, a variety of possible systems and combinations of equipment were investigated. Included in the selection considerations were the following essential requirements:

The developers of the authoring system/language must have an established track record and it must have a substantial presence in the UK market.

The system must have the ability to incorporate flexible approaches to instructional design strategies, coupled with maximum ease of use and reliability.

The CBT authoring system must be compatible with PAL videodisk equipment and be able to present computer and video images on a single screen, in colour.

The system must comply with the current policy of standardisation on MS-DOS as the operating system for microcumputers in Army training.

The Army Television Studio facilities, at ASTS, were to be used to produce video material without commercial costs and constraints.

The cost of the selected system was just over $52,000 at June 1986 prices (using an exchange rate of $1.522=£1.00). The system is illustrated in Figures 1 and 2 and consists of:

* A Zenith Z-200 microcomputer (IBM AT compatible).
* A Pioneer Laservision videodisk player (PAL).
* A high quality dot-matrix printer.
* The Tencore authoring language.
* A PLUTO graphics image digitiser and peripherals.

Figure 1 INTERACTIVE WORKSTATION

Figure 2 PLUTO GRAPHICS SYSTEM

Equipment Acceptance

A number of problems arose during the installation and acceptance trials of the overall system. These were mainly the result of the procurement procedures in force at the time. The problems encountered were far greater than anticipated and included hardware incompatibilities between the various components causing difficulties in system integration. This required extensive liaison between the system supplier and various hardware component suppliers.

PROJECT PLAN

A critical path analysis chart of the project is shown in Figure 3. The chart is only a partial representation of the project, (it does not extend to validation) and it makes a number of assumptions. This chart formed the core of the project plan and in spite of some time delays associated with procurement and system acceptance, was in general adhered to.

Project Team

An IV project is not an individual task, but requires a team. There are six main functions and areas of expertise. These required the skills embraced by:

*Project Officer.
*Subject Matter Expert.
*Training Designer.
*Systems Expert.
*Computer Programmer.
*Specialist Media Advisers.

Prior commitments required the designated team members (five principal members during the critical design period, with three others available on an ad-hoc basis for advice on video technology, TV studio capabilities, quality control and learning styles) to contribute to the project concurrently with their other tasks/projects. This staffing level was never realised and the project was essentially conducted with two officers. The man-days available were less than had been forecast and this compounded the delays experienced in procurement and acceptance.

PROJECT MANAGEMENT

The approach adopted required a consideration of the project life cycle, the guidelines to be adopted, quality and progress checks, and modification reviews. These mechanisms and their relationships involved:

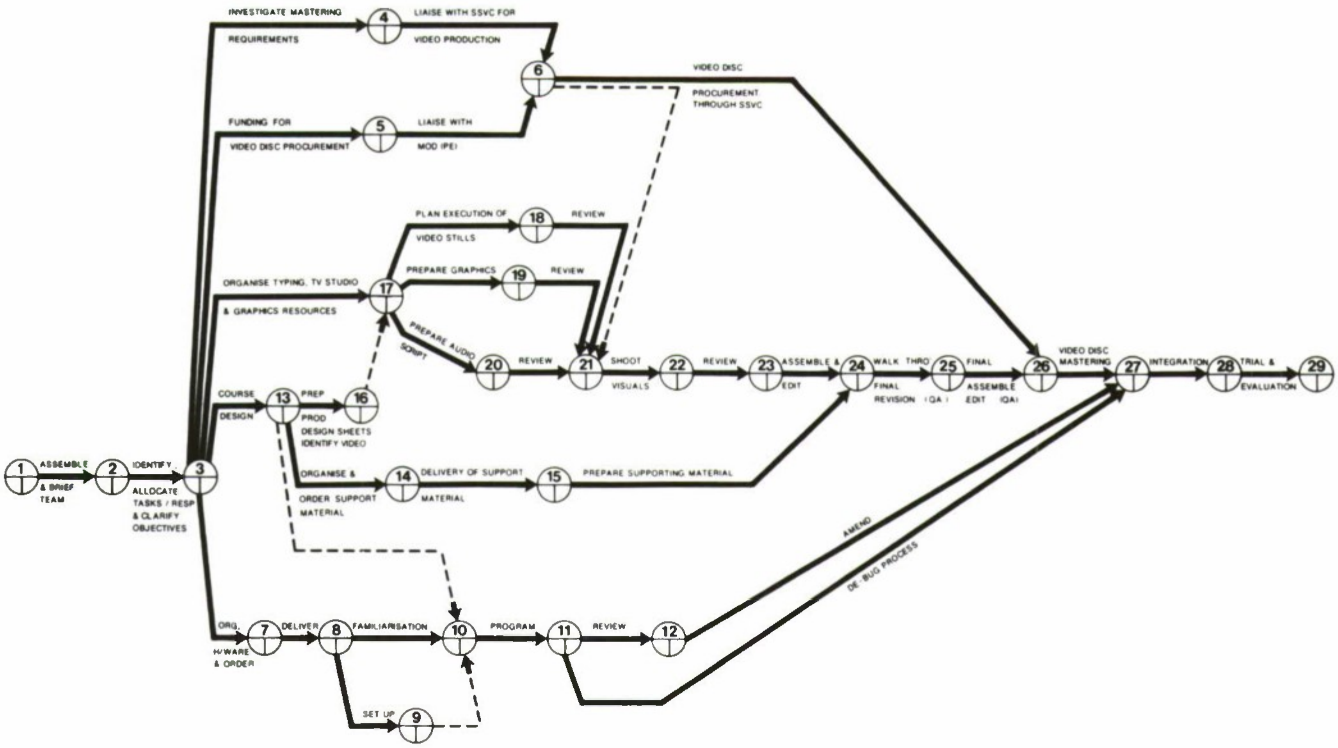


Figure 3 PROJECT MANAGEMENT CHART

The Project Life Cycle

Central to the project, this was taken to include all elements from tasking to system evaluation and initial courseware trial. The cycle considered the project to be decomposed into identifiable activities which could be evaluated. This facet of project control was the core of all the other elements within the concept of project management.

Guidelines Adopted

There were three areas for which guidelines were determined. These encompassed the tasks and activities, the procedures, and the project documentation.

The tasks and activities guide detailed what had to be done and the relationships between these activities.

The procedures guide described how the activities would be performed.

The documentation guide prescribed the form in which the progress and completion of each element of the project would be recorded.

Quality & Progress Checks

Whilst not established as a formal mechanism, checks were made internally by the project team with verbal reporting. In view of the R&D nature of the project, this was deemed to be acceptable at the time but in practice, the project would have benefited from a more formalised procedure, had resources allowed.

Modification Control

This was an activity to monitor changes in the course development and the consistent interpretation of the design by team members throughout the project life cycle. Because of changes to the team composition, and the need to accommodate other priorities, there was a lack of coherency in this procedure.

IMPLEMENTATION AND POST-IMPLEMENTATION

Terms of Reference

The terms of reference for the IV project were:

* To extend R&D on the use of IV in Army training.
* To assess the problems in the processes and production of IV courseware.
* To recommend a course of action for the Army in the use of IV in training.

Subject Identification

It was considered desirable to select a subject currently taught in an Arms School (giving Army-wide utility). The practicality of working away from the unit for protracted periods, however, ruled out any School but ASTS. Consideration of the Training Development Courses run at ASTS identified Course Design (2) as a suitable area for the project, and within this area Instructional Analysis was selected, since it incorporated task simulation as part of the course and current experience suggests that simulation is appropriate for CBT/IV. The main criteria for subject selection included a consideration of the following indicators should be:

* Visualization of tasks formed part of the course.
* Training courses were to be modularised.
* A need existed for courses to be more flexible.
* Repetition of courses.
* Trainee starting levels in knowledge and ability varied considerably.

Potentially, there would be a secondary advantage in that the material would be capable of extension into distance learning concepts, such as for the Managers of Training, both in the Regular and Territorial Army, at their parent Units.

Subject Content

The content of the module consisted of:

Context Setting. Since the module was to be used in a stand-alone setting it was necessary to provide a feel as to where IA fitted into the overall Systems Approach to Training (SAT), model shown in Figures 4 and 5.

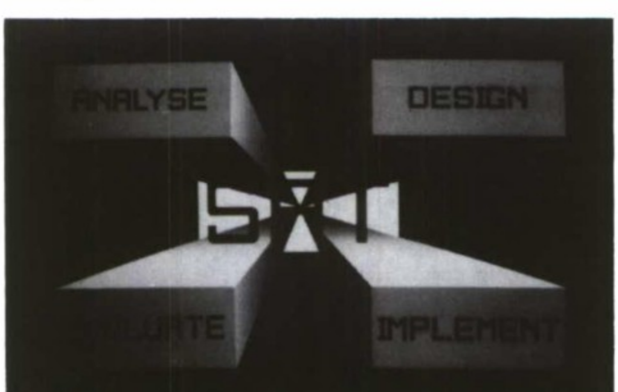


Figure 4 DESIGN AS A COMPONENT OF SAT

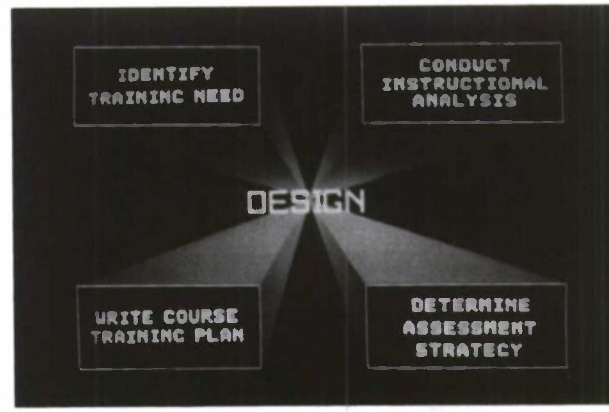


Figure 5 INSTRUCTIONAL ANALYSIS AS A COMPONENT OF DESIGN

Definition of Terms. This introduced the technical terms the user must understand to make effective use of the module. Minimal prior knowledge was assumed.

Process Demonstration. This consisted of a "walk through" of a simplified task, using reference materials, task observation, identification of task components, and the construction of a scalar. The two demonstrations were of familiar tasks, making toast and Cardio-Pulmonary Resuscitation (CPR). Within these demonstrations, good and bad points could be identified and any unrecognised assumptions or inconsistencies highlighted. The importance of the walk-through was particularly apparent in the second demonstration, CPR, when it was noted that the SME used two different hand grips for compressing the chest without realising it and also made assumptions concerning the patient's breathing, and the location of the carotid artery. Trivial though these might be in the context of the task selected, they do illustrate the difficulties likely to be experienced by those involved with developing training courses.

Practice Task. Having walked through the various stages of IA the whole topic was brought together through a second study. The aim being to build a model of all the components in IA and use it as a basis for assessment. This took some considerable effort and again was a significant element to the complexity mentioned earlier.

Assessment & Case Study. This was the final element of the module, and was only partially computer based. A very effective method of promoting learning is the use of syndicates. In order to retain this feature of the TD courses at ASTS, together with team working, the case study was delivered by the computer, but the work was prepared using materials which would be available on the job. The case study was then presented either to another syndicate or to the Directing Staff.

Supporting Activities

ASTS undertook a review of a series of other IV packages, including:

* National Bus Co. Crew/passenger relationships.
* Post Office. Inter-personnel skills for supervisors.
* British Gas. Systems Approach to Training.
* Interactive Information Systems. Interviewing.

These were found to be of variable instructional quality, but generally of a straightforward and principally linear in form. The ASTS programme was of greater complexity with more effort put into the remedial instruction where students wrongly answer the questions put.

A Basis for Design

"Begin at the beginning and go on to till you come to the end: then stop." said the King to the White Rabbit (as Lewis Carroll would have it). This would seem to be a reasonable way to proceed, and so it was in the past, but today the Training Organisation uses tools such as computers, video, and graphics all linked together. Such an arrangement is a complex system and the established ways of thinking - of managing things - is no longer competent to cope. The need now is to manage the complexity in training. This is a reflection of the increase in the complexity of operational roles, developing technology, and the increasing pressure on scarce resources.

Looking at examples of CBT - and IV is an enhanced CBT system - many of these do not measure up to expectations, this was referred to earlier. One of the problems, and there are many, is how to design a truly interactive program and not just one that is essentially linear, with a minimum degree of branching. This branching, if extensive, is where further complexity (in the course design) can arise. There is no doubt that a well designed branching programme is superior to a linear programme. This introduces variety and variety is an integral part of any effective training situation. Variety is a measure of complexity, it is defined as the number of achieve. Show me an interactive linear programme, and I will show you a denatured entity! Trying to specify all possible pathways and conditions in any program design that is non-trivial, is a brief that God himself could do nothing with! Design in the past has depended heavily upon flowcharting, as a method of representation of the sequence of operations.

This alone is totally inadequate to cope with the degree of complexity which effective CBT can imply. One step forward, and it is only that, is to employ a methodology which includes an interpretation of a Structured Design Method (SDM), in addition to the more usual tools. This method of representation will be referred to later. It is not a panacea but does allow an increased degree of flexibility and interaction to be accommodated. From this, the design sheets showing all the visual elements, together with their associated audio and text, as appropriate, were developed.

The design of interactive branching it must be more than just a re-routing through previous material. It must, for example, provide options for such activities as:

Help - related to the position from which it was invoked.

Directions - the user must not be left in the position where the next step is a matter of guesswork.

Glossary - the various technical terms should be always available for reference. degree of flexibility and interaction to be accommodated. From this, the design sheets showing all the visual elements, together with their associated audio and text, as appropriate, were developed.

Suspension - this should enable the user to temporarily halt activities and return to the same section, at some later time, if desired.

Review - depending upon whether the user has completed the module before, this option should provide the means to review any of the module elements.

Remedial - this must include provision for a variety of strategies. These should include new approaches, such as using fresh video from a different perspective, different language levels, changed forms of text, and possibly alternative learning styles. If understanding was lacking before, merely repeating the same sequence may be unproductive. An option for repetition should however be available at the user's selection, since the problem could be inattention.

It may be argued that such activities should be built into the authoring system. It is the authors contention that what should be done and that which is done, often diverge. Whilst it is comparatively easy to state what should be achieved and how, in practice the achievement of this is often lacking. The glossy production is all too easy to produce.

Design & Development

The design and development of the courseware was approached by considering it from three related but separate perspectives. In outline, these consisted of:

A Program Structure Overview. This was produced using a methodology based upon an interpretation of SDM [3]. This method of representation (Figures 6 and 7) were used in the project to represent the events that would affect the trainee progressing through the course and the control functions of the courseware. There were a number of positive attributes to this approach. These included:

* Program documentation being part of the design process.

* The logical structure providing easier and more thorough testing.

* Flexibility in design and an enhanced standard of maintenance. This is because it is clear where any alterations are required and the locations can be easily identified.

* Rigour enables ambiguities and errors in specification to be identified early, rather than at the trial stage.

A Flow Chart. This showed the overall program structure and trainee interactions with the course. This also represented how, when and what material the trainee would be presented with.

Screen Layout Sheets. These specified in detail the precise information that would be presented and how (positioning, colour, style), options available and the control functions to be provided [4].

There is nothing new in these techniques, the essence is to bring all of these aspects into a logical and coherent entity. Each view gives only a partial description, each describes only one aspect of the process, together they provide a comprehensive picture of the authoring requirements. None of these views are created in isolation, each requires user involvement.

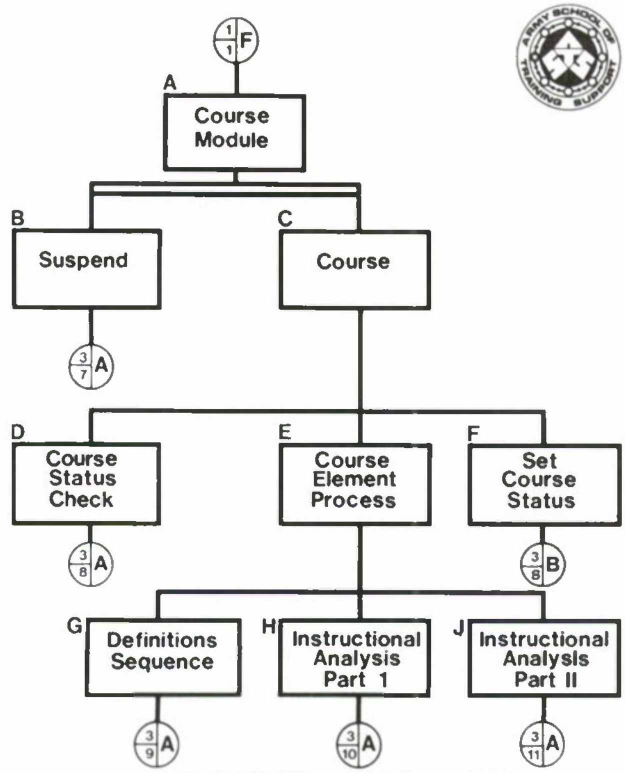


Figure 6 INSTRUCTIONAL ANALYSIS PROGRAM STRUCTURE

Courseware Style

The philosophy of CBT is one which proclaims that students are trained individually in response to their particular needs, whilse allowing a measure of trainee control. The potential of this for the accommodation of management, the monitoring of performance and matching trainee needs to the training courseware is tremendous.

The difficulties of realising such potential, however include those of an increasing burden upon the training skills and resources available. An example might be the need to recognise and take account of a wrong answer, other than in a trivial sense, and provide a number of different views or approaches relating to the same subject matter or task. This situation has the potential to increase the complexity of the courseware design to a stage where it becomes unmanageable using existing authoring tools.

A limited study was carried out at ASTS to investigate the extent of serialist/holistic learning styles (after Pask) within the target population that would be using the IV Instructional Analysis module. The purpose of the study was to explore the application of this approach to CBT/IV. Initial results [5] indicated that there was a definite serialist trend within the broad spread of styles. This suggested a mainstream design with a serialist bias.

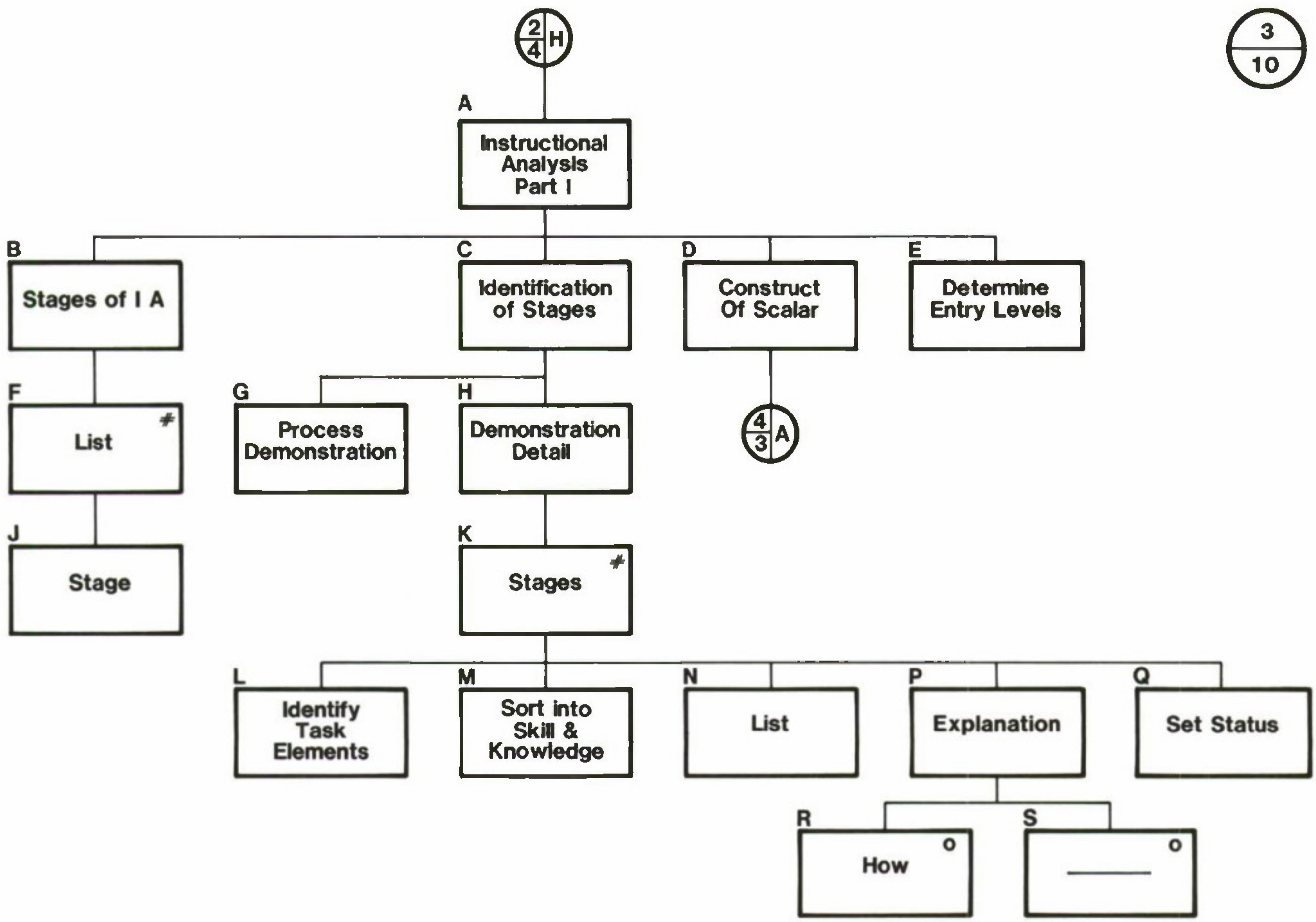


Figure 7 PROGRAM STRUCTURE-DETAIL

Video Production Requirements

Within the terms of reference, there was a requirement to investigate the feasibility of using the Army's video facilities in producing IV courseware. The equipment available at ASTS and used in the project consisted of a low-band U-matic system (ANSI Type E videotape format) using 3/4-inch cassette tapes, with the ability to record digitised graphics pictures from the PLUTO system. All source tapes were therefore U-matic, and additional material, in the form of stills, was produced using 35mm photographic slides. The master tape for disk pressing was produced outside ASTS by the Services Sound and Vision Corporation (SSVC).

Validation

To establish the effectiveness of the IV module there will be a need to implement a validation program. It is proposed to conduct a number of comparative trials within ASTS in the Winter of 1987.

CONCLUSIONS

An IV project requires the commitment of a team capable of performing six main functions. Unless an establishment is lavishly resourced, this level of effort is very difficult to sustain over a long period when there are competing demands and changing priorities.

It would seem that the training needs of the Army do not equate to the perceived needs of many commercial organisations within the UK. in particularly, the outcomes of training for the Army do not appear to be the same as the expected outcomes in the commercial world, where the considerations of marketing, image, and public relations (PR) are significant factors (eg. IBM point of sale programme in the UK). This preliminary conclusion is based upon a limited review of some of the private sector IV training programs.

The project team must exist throughout the duration of the project as a coherent unit. This is not a new proposal, it is a reinforcement of previously stated views.

The method adopted in design and development paid dividends in terms of time and reliability despite the appearance of this adding to the project overheads.

The selection of suitable topics for CBT/IV requires a re-appraisal of the implementation of the criteria advanced in the past. In many cases these may be too loose, or may not take sufficient account of operational need. Examples would include:

* Ratings of CBT/IV benefits - often subjective.
* Decentralised instruction - may be an argument for distance learning, not CBT/IV.
* Student throughput (quantity) - should also take account of quality.
* Consideration of existing or forecast on job performance.

A more selective and critical assessment by prospective users of IV would improve cost-effectiveness since time and manpower are increasingly scarce resources. It would seem, as a generalisation, that the use of IV in various forms of simulation would be the most fruitful area for exploitation, with others being the exception, rather than the rule.

The subjects for which IV may be proposed must merit the high allocation of resources and costs which IV implies. The staffing of the ASTS project indicates probable manning levels but in addition to this there are the requirements for television studio resources and availability. Such considerations indicate that the cost-effective use of IV is unlikely to lie in those areas in which training is already effective, unless other significant management factors apply.

REFERENCES

1. Interactive Video: A Review of the State of the ART and Applications. ASTS, RAEC Centre, Beaconsfield, Bucks, UK. March 1984.

2. The CAI Decision Handbook. Sue Kemner-Richardson, Joseph P Lamos & Anita S West: Denver Research Institute, University of Denver, Colorado, USA. June 1983.

3. SSADM Handbook. HM Treasury, CCTA, London, UK. August 1984.

4. Guidelines for Designing User Interface Software, Sidney L Smith & Jane N Mosier: The MITRE Corporation, Bedford, MA, USA. Publication ESD-TR-86-278, August 1986.

5. Learning Styles and Training. (in preparation), ASTS, RAEC Centre, Beaconsfield, Bucks, UK.

ABOUT THE AUTHOR

Major Tony Croucher is a Training Systems Consultant at the Army School of Training Support for the British Army. He holds an MSc in Operational Research & Systems Analysis and has 11 years experience in Army training, the last 4 of which have been at ASTS. He has been involved with the Warrior training aids concept, CBT in communications simulation and several computer related projects. He is currently the principal project officer for the application of IV to Army training.

Fault Tolerant Computational Systems

Mr. Jeffrey Pulcini
Concurrent Computer Corporation
Binghamton, NY 13901

ABSTRACT

Over the years, the use of trainers has become more vital in ensuring operational readiness. Because both training and personnel time are in short supply, the training device should be operational both when scheduled and during the entire training session. This latter requirement has become more important as long, simulated missions are increasingly utilized to insure full crew/mission training. The objective of this paper is to introduce the engineer to the concepts of availability and fault tolerance. It does so by addressing the topic in three parts.

Part one describes levels of fault tolerance and works to put bounds on the problem. This is vital since various fault tolerant concepts might include costly and unnecessary components such as uninterruptible facility power and full fault detecting software and hardware.

Part two describes example hardware and software systems that will achieve the designated levels of fault tolerance. By utilizing examples, key system elements of the hardware, as well as system and application level software can be highlighted and discussed. Each of these entities must have attributes that will map into a fault tolerant philosophy, thus determining the approach and cost of the resulting system.

Finally, part three examines some of the end user implications of fielding a fault tolerant simulation system. This section highlights such considerations as sparing, maintenance philosophy, and quality of maintenance.

INTRODUCTION

An essay was heard on Public Radio discussing the curious mentality of we Americans. The author of the essay, a recent immigrant from Russia, observed that, unlike citizens of his former country, we Americans have come to expect things to always go right. Airplanes will be on time, traffic will flow smoothly, and mistakes won't be made. If problems do occur, an immediate cry is raised to "fix the system". Fixing the system is usually translated to mean "make it more reliable".

In simulation, reliability is one of the key measures of the systems we produce. While reliability has its place, it is just a symptom of the problem. Those who conduct training on the device, the ultimate end users of the simulation systems, don't care about reliability per se, what they care about is availability. They want to train when scheduled and with the lesson and equipment they have chosen. One only needs a quick glance at the latest crop of RFPs to see the emphasis on availability.

This has caused no end of concern for those responsible for the bid and engineering of flight simulators. In our haste to meet availability goals, we have equated availability and reliability. This is not always the case. To see why, we need an understanding of the measures of reliability. Further, a fresh approach to the problem is needed. This is the concept of fault tolerance. In the next three sections, we will lay the ground work for constructing a highly available system.

CONCEPTS OF FAULT TOLERANCE

The mention of fault tolerant systems brings about a euphoric sense of well being. After all, what could be more wonderful. A system on-line... never failing... always at the command of the user. It's the kind of feeling a pilot gets just before he passes out from lack of oxygen. While in theory such a system could be constructed, it is usually impractical either technically or monetarily. Thus, physical realities dictate that even fault tolerant systems might be subject to an occasional failure. With this in mind, this author would like to offer some categories or definitions of fault tolerant systems.

Basic Definitions

The first category of a fault tolerant systems is one that provides graceful degradation. In this category, should a failure occur in the system, the user would experience a partial loss in performance, functionality or both. There is no backup hardware in the system. In reality this category breaks down into two classes.

- Low automation, low cost
- High automation, low cost

The first class, low automation-low cost, is distinguished by the fact that a failure requires either human intervention or specific programming effort to correctly respond to the failure.

The second class, high automation-low cost, is defined as a software system that would, without user intervention, detect a failure and redistribute the application so that the user would experience first a loss of performance followed by a loss of functionality.

Of the above two classes, only the low automation solution is currently achievable with commercial computation systems. The high automation solution, while receiving academic attention, is not practical with todays general purpose software systems.

The second category of fault tolerant systems is the resilient system. In this system, backup hardware is provided and the user has provided the system with a scenario to follow should a fault be detected. This

system is most akin to the low automation system above, but the cost has been moderately increased by the addition of a redundant hardware system.

Finally, the last category of fault tolerance is the fail-safe processing system. In this system, not only is all hardware found in (at least) triplicate, but every data path is as well. Further, computations and results are compared and a processor with "non conforming" results is eliminated.

Figures One through Three show basic diagrams of computer systems with some associated components. While they are not simulation systems, we can learn a lot from those who have gone before. Each diagram will be discussed in detail under the section entitled FAULT TOLERANT SYSTEMS.

While the hardware is always at the forefront of the discussion, a more pertinent question is one of software. Invariably, the purchase of hardware is not enough. Any software in the system must be able to take advantage of the fault tolerance the hardware provides. Further, there is a question of philosophy— how much is enough?

Fault Tolerant Philosophy

While this subject might sound esoteric, it in fact is the crux of the problem. For example, if the function being performed is one that would cause the loss of life if a fault went undetected, then only the most sophisticated system will be acceptable. At the other extreme, a fault resulting in a nuisance condition may not be worthy of consideration, let alone having to expend additional time or monies to detect and correct it.

In reality, most conditions we have to deal with in the simulation environment lie squarely in the middle. What is at stake is loss of valuable training time, negative training, and wasted manpower and the dollars associated with this. Thus, it becomes important for us to establish a "threshold of pain" for this loss of training. To a certain extent, this has been done with the recent move toward CLS. The RFPs are now beginning to specify minimum availabilities. From these, we can begin to do "failure analysis". This analysis allows us to look at the problem and determines which components, if they fail, will preclude mission success.

A simple example can help show this concept and introduce some valuable terms. If we require an automobile to complete a mission, we can examine the components of the car to see how critical each one is to our success. The first component we examine is the clock. If the clock fails, is will have no effect on our mission. Therefore, making a more reliable clock, or providing redundancy will only drive up the cost and not enhance our performance.

On the other hand, if a tire fails, this would be a failure that would effect our ability to complete the mission. This failure, however, need not be catastrophic. If we choose to have the redundancy of a spare tire, then our mission might be delayed, but not canceled.

Finally, we might examine the engine. If the crankshaft in the engine failed, this would jeopardize the completion of our mission. True, we might carry a spare crankshaft, but the tools and time necessary to affect the repair would be prohibitive.

This all seems a common sense approach to making implementation choices with respect to a fault tolerant system. Unfortunately, many of the decisions made in a complex weapons system simulator would not be cut and dry. It would help if we had some objective measures to help make these decisions. These measures are: Mean Time Between Failures (MTBF), Mean Time Between Critical Failures (MTBCF), and Mean Time To Repair (MTTR). To assist in analyzing the effect each failure would have on our mission, there is a method known as a Failure Modes, Effects, and Criticality Analysis (FMECA).

While it is beyond the scope of this paper to discuss the complete philosophy and implementation of each of these measures, a brief description will show how, when taken *together as indicators*, they can be used to help in the decision process. Let's look at a simplistic definition and the interaction for each of these measures.

FAILURE METRICS

Mean Time Between Failures, MTBF, is the statistical measure of failure frequency. The operative word in that sentence is statistical. Take for example a computer circuit card. MTBF is found by taking the failure rates of each component on the card, under certain conditions, adding them all up, and taking the reciprocal. Taking three components that each had a failure rate of once every 1200 hours, the MTBF of a card consisting of those three components would be 400 hours. A pretty simple system. The problem is, that there are a number of variables that can not only change MTBF, but could change the actual failure rate. These variables are:

- — operating conditions
- — the actual operational role the assembly plays.
- — the critical role of the components

The MTBF of a component might be measured at 50°C. If the operation is actually at 20°C, the the MTBF is likely to go up. The second condition, the actual role the assembly plays, is a bit more difficult to see. Let's suppose that a card is placed in a system and, in that system, the card is rarely used. This may implies that current flow through certain components is reduced, heat generated is less and thus, the component lasts longer. Once again, the MTBF should go up.

MTBF may fall short in another area. Not all components may be critical to the operation of the device. Some components may be in a circuit or device to compensate for extreme conditions or to provide optional functionality. In such cases, failure of such components, while constituting a technical failure, may not even be noticed. Back to our example of a car, the clock had components that contributed to lowering the MTBF of the vehicle, but in reality, didn't affect its operational capability. Recognizing this dilemma, the measure MTBCF, Mean Time Between Critical Failures, was created. It is this measure that will helps us determine the availability of the device.

Of course, we now have the additional burden of determining what's critical. To assist in this determination is a method known as Failure Modes, Effects and Criticality Analysis, or FMECA. A simplistic picture of a FMECA is a top down analysis of our device. Returning to our car example, we would break down the car into it's major systems (body, engine, drive train, interior, etc.). We would then determine what system, if failed, would jeopardize the mission. At this high level, engine and drive train might be singled out. We would continue to break down these systems to their subsystems and repeat the process, not only determining other "show stoppers", but assigning a "measure of effect" to those critical items.

To illustrate, imagine going through the drive train subsystem of our car and identifying the components such as the differential, axle, wheel and tire. A failure of the differential or axle would be catastrophic and receive an "effect measure" of ten (the highest). A failure of a tire would only receive a measure of two (next to the lowest). Why only two? Because at this point we could make a couple of decisions. First, a tire while having a high failure rate (low MTBF), is easy to replace. The Mean Time To Repair (MTTR) is low. The Second decision is that the tire is sufficiently easy and inexpensive to spare. Thus, in terms of impact to the mission, the effect is also low.

So, if we continue in this fashion and do a FMECA on every assembly in the system, we could develop an MTBCF to correspond with an MTBF.

At this juncture, we have laid the ground work for further examinations of fault tolerant systems. We should realize that, while these measures can aid us significantly in guiding our design decisions, they should not become goals in themselves.

FAULT TOLERANT SYSTEMS

Turning our attention back to Figures One through Three, we can now begin to describe how concepts, philosophy and the measurement criteria can be combined to give us devices capable of meeting our training needs.

Graceful Degradation

Figure one is the most basic fault tolerant system and one that meets the requirements for graceful degradation. In this system, multiple processors and multiple peripherals share the processing and I/O load to create a system that meets the total needs of the mission. This example shows the first criteria of a fault tolerant system, design from the beginning.

As engineers, we are taught to state the problem and, in doing so, we place bounds on the problem. We could extend this concept to high availability systems by performing a mental FMECA. By asking which functions performed are critical to mission success and defining these up front, we can begin to apportion the problem among the various pieces of hardware.

If we use as an example a basic flight trainer, we might place the flight package as a critical item. On the other hand, some of the advanced training functions such as record playback or demonstrations might fall to the reduced requirements category.

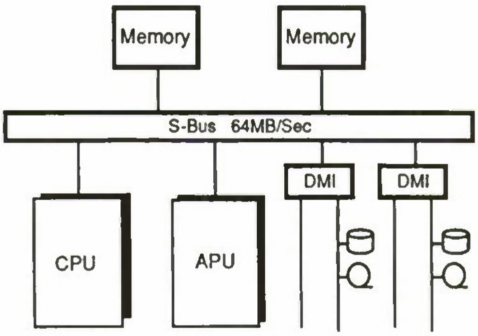


Figure One
Low Cost, Fault Tolerant Solution

Once the functions have been established as primary and secondary, we can then select a computer system that will accommodate them. We have duplicated the components that are most likely to fail and sized them accordingly. A single large processor might be replaced by two smaller processors. For I/O, two smaller disks could replace a single large disk while the cockpit I/O system may also have duplicated components. The memory system and power system are not duplicated. Power supplies have proven to be very reliable components and the memory system has error correction and detection as well as error logging capabilities thus facilitating failure trend analysis.

In this system, the philosophy is one of work around. If one of the disks fail, data would continue to be recorded on a the other media. If a processor fails, the other processor could be assigned the job of processing the primary functions while leaving undone the secondary functions or, perhaps, continue processing the full software suite, but at a reduced frame rate. Should a hard failure in the memories occur, todays operating systems can usually work around that failed segment.

No matter how we choose to respond to the failure, the object of the game is *graceful degradation*. We may not have all the bells and whistles, but training can continue. Some questions that have to be answered with this scenario are:

- how will switch over be accomplished
- how long will it take
- how noticeable will it be
- what software provisions have to be made to facilitate a down grading.

The answer to the first three questions will be predicated largely on the computational system chosen. Systems that have full software capabilities to reassign I/O and move processes from one processor to another with the minimum of fuss will obviously do better than those where the user has to work around a non-responsive operating system.

The fourth question can be answered only in light of the design work done before hand. In a system attempting a graceful degradation, some sort of checkpointing is normally utilized to create a "resynchronization" or "restart" point. When this design is used, it tends to have a trickle down effect on the rest of the software. In a simulator, the designer can take advantage of "natural" checkpointing and restart software already existing in the form of record-playback, freeze flags, and frame starts.

The Resilient System

Figure Two is the next logical step into fault tolerance, the resilient system. In this system, the hardware is duplicated and the systems are interconnected via LAN or a high performance memory link. Should a primary system fail to provide "I'm OK" messages to a second system, the second system seizes control and assumes the processing responsibilities. Figure Two shows a simplified configuration of a system used by G-Tech Inc. of Rhode Island. G-Tech provides very successful gaming systems used in state run lotteries.

While this is not simulation, the environment is in many ways more demanding. The number of terminals served by the polled lines can be up to 12000. The computer system is designed to acquire data from groups of polled lines, decrypt the data, compute the result, verify the result with the second system, log the completed transaction to disk, encrypt the data prior to transmission and send the results back down the line. The systems produce up to 45,000 transactions/minute (once every 1.3 ms).

The technical details, while impressive, almost pale against the political and monetary realities. These systems deal in peoples money, and lots of it. Each player must be assured his ticket is unique and his wager recorded. System integrity is paramount. Both the player and the state demand the system always be available, though for different reasons. For the state, this is very big business. An unavailable system means lost revenue. For the player, if the system fails when the lottery has a particularly large prize, he can't take that "once in a lifetime chance" and feels cheated. To insure accuracy and availability, most states assess penalties for down time. If a system should fail, penalties could exceed $100,000 per hour.

So, what's the track record of this system? G-Tech has documented availability ranging from an all time low of 99.91% to a high of 99.99% based on a 17 hour day, 365 days per year. The systems were unavailable between 1/2 and 5 1/2 hours per year. Of that, only 25%, *between 10 minutes to 90 minutes per year*, were directly attributable to system hardware and software problems. The rest of the down time was operator induced.

As a post script to this discussion, G-Tech has heretofore operated with the philosophy that any detected failure was cause for an immediate switch over to the backup system. As of late, they have conducted experiments where processors in a system were successively failed. In each case, the remaining processors picked up the load and continued to support the wagering. In essence, the low cost approach to fault tolerance, graceful degradation, has been validated.

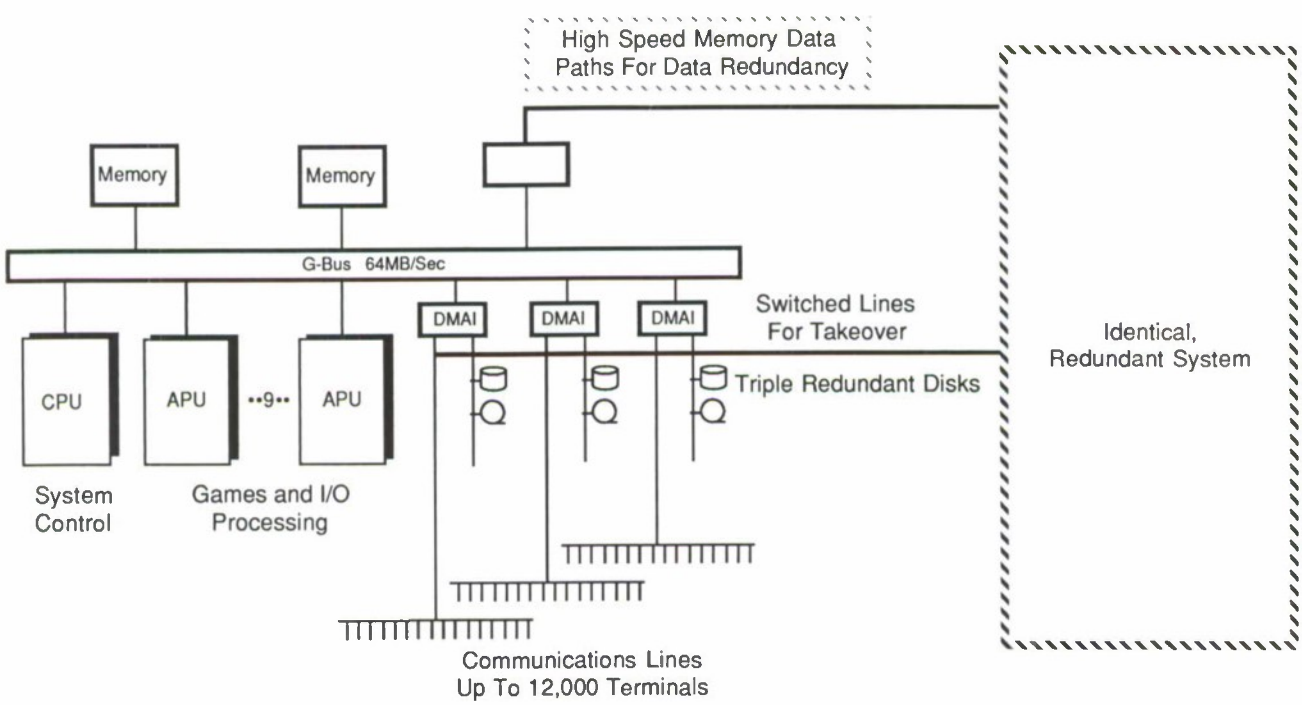


Figure Two
Fully Redundant, Fault Tolerant Solution
Courtesy G-Tech Corporation

Fail-Safe Processing

The final example of fault tolerant computer systems is the fail-safe processing system. Figure three is a simplified block diagram of the computational system that supports the flight controls of the F-16. An excellent paper was published and presented at the 1983 NAECON conference from which this block diagram was derived.[1] The essence of fail safe computing is shown in this diagram.

Multiple inputs are sent to multiple processing systems which compute and compare their results with each other and then deliver the results through multiple output channels. Not shown in the diagram is the work being done in the software. As various sensor inputs might fail or be battle damaged, the control laws had to be reconfigured to permit continued satisfactory flight control. In addition, the software was charged with monitoring the health of its processor and the health of other processors and subsystems.

As you can see, software plays a very important role. What, however, precludes a common software failure? This type of generic failure is handled in a number of ways. For the F-16 flight control system, an Independent Back-up Unit, IBU, was provided. This was a small analog unit that provided simple control which allowed the aircraft to be flown home in the event of a catastrophic flight control system failure.

In a system that could not be so easily controlled, for example the space shuttle, the functionality of the software system was recoded by a separate set of programmers independent of the primary software team. Thus, the possibility of the same coding error bringing all the processors down simultaneously (in the case of the shuttle, a five computer system), was minimized.

In essence, the fail-safe system is the extreme of the resilient system. Additional hardware and redundant data paths are a factor, but the real separation occurs in the software effort to assure total reliability of the hardware and responsiveness in the event of a failure.

Fault Tolerant Systems Summary

From the previous section, the biggest hurdle we must overcome is to begin to think in terms of high availability, fault tolerant systems. Fault tolerant computing is achievable with todays computational systems. The key word, however, is system. The system chosen must have both hardware and software capabilities on which we can build. Having chosen a system, then an *up front* design effort is prerequisite for success.

Finally, in the previous sections, we have dutifully focused on purely a technical approach to the question of fault tolerance level. In reality, there are many other factors which must be considered when fielding a fault tolerant system. The basis for these factors were discussed earlier in the sections on Failure Metrics and Fault Tolerant Concepts. The final section of this paper integrates the knowledge we have gained.

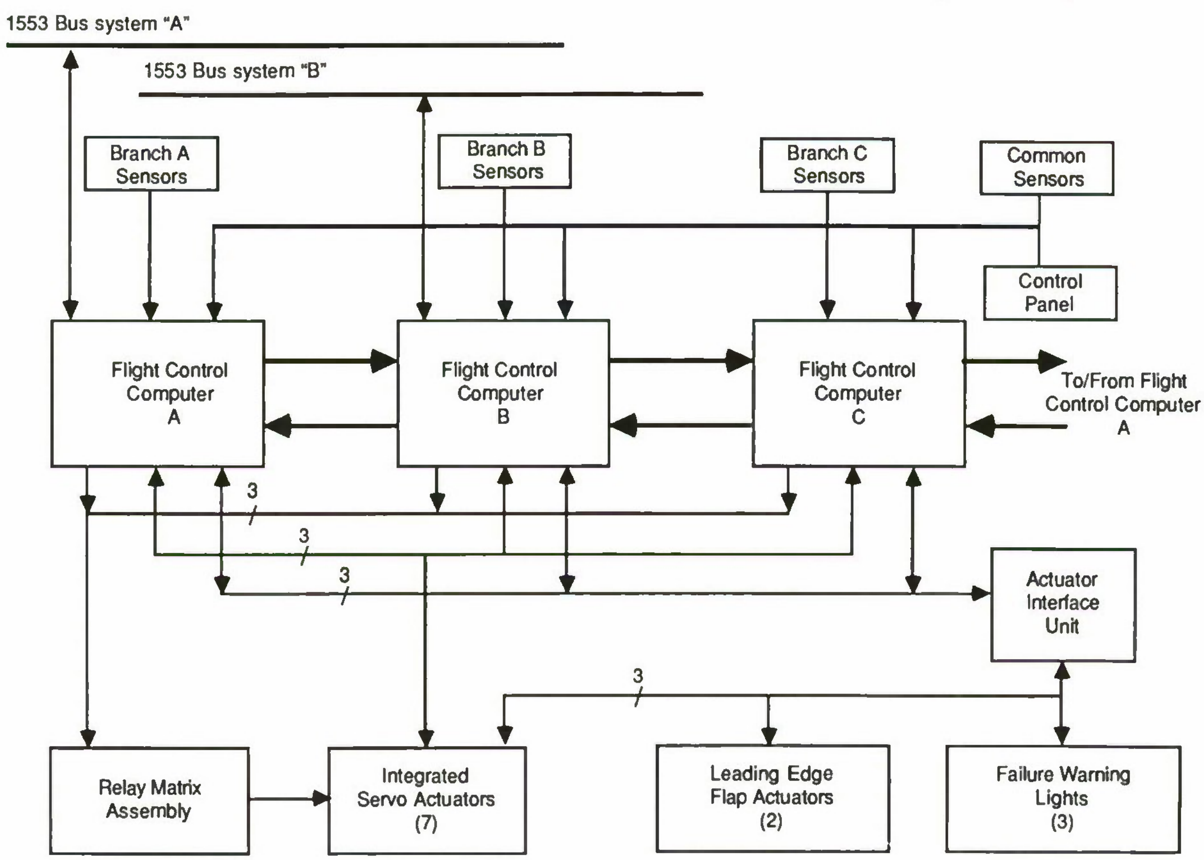


Figure Three
Fail-Safe, Fault Tolerant Solution

BUILDING AND FIELDING HIGH AVAILABILITY SYSTEMS

When looking to build a high availability system, we are performing a balancing act. This isn't new. As engineers, we do this every day, making decisions and conducting trade-offs within our designs. What makes the process of building a fault tolerant system so difficult is the number of people which must be involved and the amicable relationship and team work that must exist between them.

To build and field a fault tolerant system, this author would suggest three simple rules:

— The team should consist of at least the following representatives from both the manufacturer and the customer:

Design engineer
Production engineer
Reliability Engineer
Quality Assurance
Customer Service
Program Management
Logistics
the hardware/software vendor (if applicable)

— There is neither a One Man Majority nor Minority

— There is only one objective: Field a device that meets the availability needs of the user at the lowest possible life cycle cost.

A brief look at the each rule will show why each rule is suggested.

The first rule is designed to get all the right people involved when it will do the most good, up front. The Defense Systems Management College has demonstrated that "Decisions affecting 70% of the life cycle costs are made by the end of concept exploration, and decisions affecting 85% of the life cycle costs are made before full scale engineering development begins".[2] Further, no one organization can hope to produce a system that is available, cost effective, and meets the end users needs. Checks and balances are needed.

If we allow, even the best qualified, most well intentioned design engineer to be solely responsible for deciding which functions are of primary importance to the user and which are secondary, its a guarantee he'll guess wrong in some of the cases. In this example, the end user provides balance by giving the operational insight necessary for the decision process.

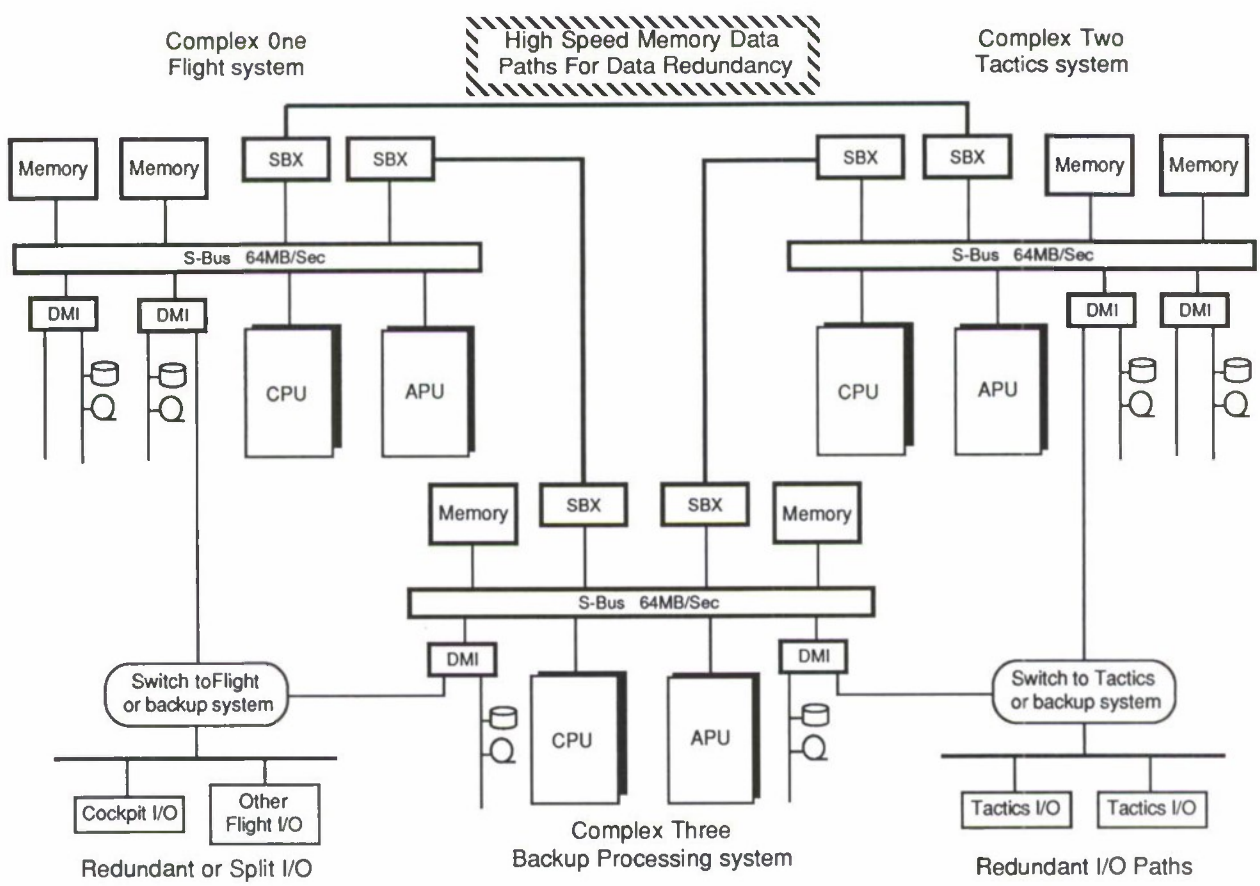


Figure Four
Fully Redundant, Fault Tolerant WST

Sadly, having a representative on the team doesn't guarantee representation. Rule Two, no One Man Majorities or Minorities, attempts to moderate strong personalities and biases within the group. In a recent interview, Mr. Willis J. Willoughby, the Navy's director of reliability, maintainability and quality assurance, was asked why he emphasized design and manufacturing. He responded saying, " Our reliability and quality assurance guys are only reporters. They tell us what happened. I want to do away with the word 'reliability,' because it's a subset, a cult. ...Ultimately, the reliability and quality guys should be working right along side the design and production guys, or, better yet, the design and production guys should be responsible for reliability and quality assurance."[3]

You can readily see the danger. If all of our energies were focused on driving MTBF to some maximum, one of the most effective ways is to reduce the number of components. Yet, our fault tolerant examples showed the most available systems were the ones where the number of components at least doubled. On the other hand, if we totally ignore MTBF and, therefore, the impact on life cycle cost, the system we produce may well be too costly to support.

The net of this is that each decision or trade off that is made has got to be a team decision that is fully justifiable in business terms. Not just money, but mission as well.

Invariably, personalities, self interest, or just plain honest disagreements may get the better of our team members. It is at this time, rule three is applied. There is one objective, get the job done for the customer. Disagreements may occur, but each member must then agree to disagree and continue on with the job.

One final example is in order. Figure Four illustrates how a computer system might be configured for a small Weapon Systems Trainer. Computer Complex One is configured to handle the chores of flight, navigation and training systems. It contains multiple processors on a single memory system designed to backup one another by load shedding or reallocating the simulation tasks into each processors spare time. Dual I/O channels are employed to allow for backup recording capability during training.

Computer Complex Two is configured to support the Tactics portion of the WST with the same level of redundancy. Each system is interconnected via the memory system to allow the sharing of data as well as providing an alternate I/O path to the cockpit.

Finally, a third system is provided and linked to Complex One and Two's memory and I/O system. Complex Three has the full simulation load on line and constantly receives updated results from each of the two primary systems. Should either system experience a catastrophic failure, Complex Three will activate the appropriate set of simulation tasks and the simulation will continue with only minor interruption.

CONCLUSION

Providing high availability systems is potentially a costly and time consuming process. The question must be asked, is there any support for this from the DoD? The answer is clearly yes. For example, the Air Force has embarked on a program called R&M 2000. While it has emphasis on combat equipment acquisitions, the effects are being felt in the simulation community. Simply stated, the Air Force wants equipment that will fly mission after mission, reliably.

On a broader scale, Secretary of Defense, Caspar W Weinberger in his Annual Report to the Congress, Fiscal Year 1987 has said, "I have said yes to ... demanding that the reliability and maintainability of our weapons systems be considered equal to cost, schedule, and performance during the acquisition process."

REFERENCES

[1] Arabian, Alan M., Naumann, Erwin A., and Swihart, Donald E.: "AFTI/F-16 Digital Flight Control Computer Design", IEEE Paper Number 0547-3578/83/0000-1426, Proceedings of the 1983 IEEE National Aerospace and Electronics Conference (NAECON)

[2] Reliability and Maintainability Principles brochure,Pg 5, HQ USAF/LE-RD, First Edition, October, 1985

[3] Willoughby's Way, James Kitfield, MILITATY LOGISTICS FORUM, pp 30-36, June, 1987

ABOUT THE AUTHOR

Mr. Jeffrey Pulcini is a Member of the Technical Staff with Concurrent Computer Corporation. His current duties involve design of system solutions to complex real-time applications. Mr. Pulcini has previously held positions of Product Manager for Advanced Systems and was the initial architect for Concurrent's 3200MPS series of Parallel Processors. Mr. Pulcini holds a BS in Aeronautical Science and has authored numerous papers on computer performance, architecture, and selection.

A TIGHTLY COUPLED DISTRIBUTED SYSTEM FOR FLIGHT SIMULATORS

John R. Bocskor
Robert J. Cichon, Sr.
Gould Inc., Information Systems
Computer Systems Division
6901 West Sunrise Boulevard
Ft. Lauderdale, Florida 33313-4499

ABSTRACT

To attain the realism necessary for simulation today, higher and higher system fidelity is required. Initially, all simulation software was controlled and executed on a monolithic processor that had to complete execution of all software modules within a specified time frame. As simulation requirements increased, it became evident that portions of the simulation software could be executed in parallel. To meet the requirements for increased fidelity in simulators being designed today, the software has been divided into several cooperating modules. These modules generally load and execute in a number of computers connected by a portion of common physical memory referred to as shared memory. These conventional shared memory systems are typically used in cases where true parallel processing takes place. The shared memory system allows for high-speed coupling of computers which in turn allows higher frame rates thus better fidelity. A new method of tightly coupling multiple computer systems without the inherent deficiencies of conventional shared memory was needed. In addition, a new hardware implementation that utilizes gate array technology and a means of controlling such a system from a designated Host System are required.

INTRODUCTION

In the four sections that follow--Shared Memory Hardware, Computer System Hardware, Software Control, and Diagnostic Software--the authors will discuss the problems encountered and the solutions arrived at in developing a tightly coupled, distributed system for flight simulators.

In brief, the problems were three-fold: 1)getting beyond the conventional shared memory approach, 2), "shrinking" the footprint of the system while enhancing its compute power, and 3), reducing the life-cycle cost.

SHARED MEMORY HARDWARE

A. Conventional Shared Memory
(see figure 1)

With this approach there is typically a Shared Memory Interface Controller connected to each node's main bus. In turn, each node is cabled to the Shared Memory Controller located in the shared memory chassis. The Shared Memory Controller actually arbitrates access to the shared memory bus on a cyclic rotating priority basis. Propagation delays on reads and writes of data in the shared memory area occur through each layer of hardware in this type of system. As distance between the shared memory chassis and the nodes is increased then additional propagation delays occur due to line delays.

Compute nodes are granted access to the Shared Memory Bus by order of priority. At the time of the start of the grant cycle only those nodes currently requesting shared data are serviced. Any other requests to the Shared Memory Bus have to wait until the next grant cycle, even though a higher priority node may be requesting the Shared Memory Bus. This type of arbitration is typical for most shared memory systems and ensures all nodes get access to the Shared Memory Bus.

In a conventional shared memory system all nodes connected to the Shared Memory Bus are contending for the same physical module of shared memory. In a heavily loaded memory intensive system the nodes are waiting for memory a high percentage of time. All nodes in the system are accessing this module during reads and writes of shared memory data. Contention could be reduced if the shared memory bandwidth was high enough to handle the aggregate demand of all users. These high bandwidth memories are not economically viable on todays computer systems.

Conventional shared memory systems are housed in a separate shared memory chassis and require a separate power source in addition to the other power supplies in the computer system. In some cases the shared memory chassis and power supplies can not be configured in the same cabinetry as the computer system. An additional cabinet has to be added to house the shared memory system. Spares requirements and system reliability of a conventional shared memory system are affected because the power source, the chassis and the shared memory module are typically uncommon with the rest of the computer systems hardware.

A sophisticated central clocking system is required to synchronize all nodes connected to the conventional shared memory system. A master clock is used to synchronize all the nodes' main busses as well as the Shared Memory Bus.

CONVENTIONAL SHARED MEMORY APPROACH

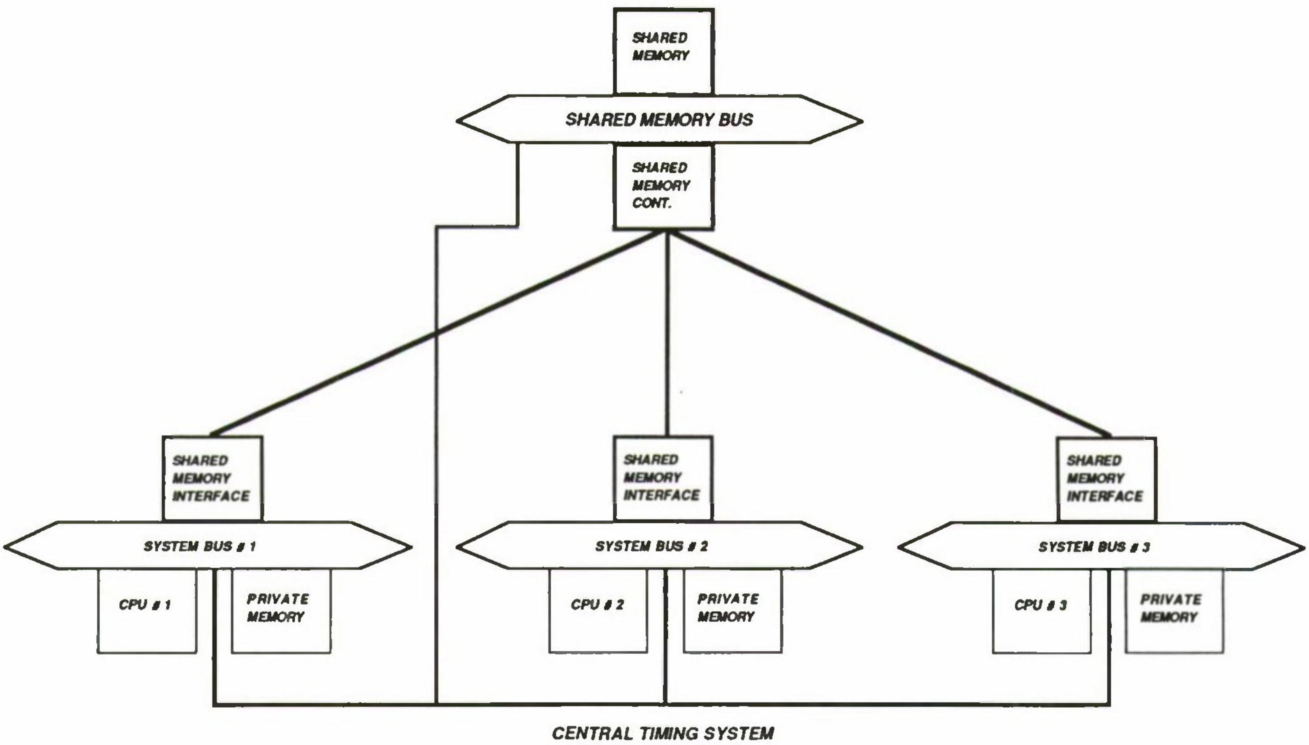


FIGURE 1

This clocking system is necessary because each node in the system sees and treats the shared memory module as an extension of its own main memory with full access privileges.

The conventional shared memory approach lends itself to many single points of failure. If a problem occurs with the shared memory power supplies, shared memory chassis, shared memory module, or the shared memory controller then, in effect all shared memory operations with all connected nodes are lost. New implementations of intercomputer shared memory systems must solve such problems.

B. <u>Dual Ported Shared Memory Approach</u> (see figure 2)

All nodes are connected by a high speed 26 megabyte (MB)/sec intercomputer datalink. The intercomputer datalink consists of 32 bits of data, 24 bits of address and control lines. Only write data within specified address regions is transmitted out onto the intercomputer datalink. Using 40 foot cables, write transactions can occur on the intercomputer datalink every 150 nanoseconds (ns), a rate which decreases to 300ns using 80 foot cables. The intercomputer datalink is not accessed by read instructions. Each node in the system maintains its own version of the common database; therefore, all data reads are local to each node's main memory. The key ingrediant to the success of the new shared memory approach is the fact that all read traffic has been removed from the shared memory system. Typical simulation code accesses the variable data set located in shared memory at a high rate per frame. There are very few writes into the variable data set but, many reads of the common data throughout each frame. The fact that the read traffic has been removed from the shared memory system has increased overall effective bandwidth of the computer system.

High speed dual ported integrated memory modules (DPIMM) are the key to the tightly coupled shared memory approach. DPIMM's are available in 2MB and 8MB sizes. The memory modules are internally two-way interleaved and can be externally interleaved for effective 4-way interleaving. Interleaving allows multiple memory accesses to occur before a memory busy signal is raised on the node. Port 0 (node main bus interface) operates a 26 MB/sec and port 1 (Inter-computer datalink interface) operates at 26 MB/sec using 40 foot cables and

DUAL PORTED MEMORY APPROACH

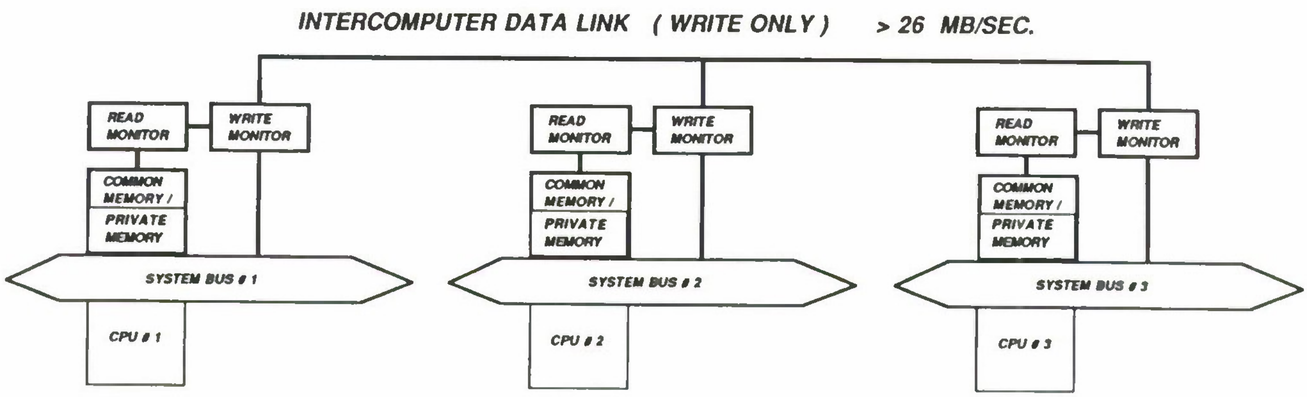


FIGURE 2

13MB/sec using 80 foot cables. Each dual ported memory module has a 4 deep request latch on port 0 which allows four memory requests to occur before the memory busy lines are raised. Port 1 has a 2 deep request latch.

Simultaneous accesses to the intercomputer datalink are arbitrated on a cyclic rotating priority basis. All requesting nodes are assured of getting access to the intercomputer datalink with a worst case wait of (8) clock cycles. This case would only occur if all nine nodes request the intercomputer datalink simultaneously. There are no cycles skipped if the requesting nodes are out of priority order (node 0 - 8). In the case of simultaneous requests, then node 0 is the highest priority and node 8 is the lowest priority during that grant cycle. A node is granted access to the intercomputer datalink every clock tic during the grant cycle by order of priority.

Write Monitor - The Write Monitor senses each node's main bus for address ranges to be transmitted on the intercomputer datalink. The Write Monitor senses writes on the node's main bus within a preselected address range. The address range is dynamically controlled by software and the Write Monitor transmits the data and address on to the intercomputer datalink only if a write occurs within the preselected address range. The Write Monitor contains a first-in first-out (FIFO) buffer that can queue up to 64 intercomputer datalink write requests. This allows the local node to continue processing and not wait for the grant to the intercomputer datalink. The Write Monitor does not occupy space in the node's main card cage. It connects to the system on the back side of the main bus. Each node in this dual ported shared memory system can be set to monitor a different set of shared address ranges by initializing each node's write control register to a different value.

Read Monitor - The Read Monitor senses the intercomputer datalink for shared data that falls within a preselected address range. The address range is dynamically controlled by software and the Read Monitor writes the data into the second port of the Dual Ported memory module only if the address is within the preselected address range. Each node in the system has its own copy or a subset of the shared data based on the Read Monitor address range. The amount of the shared data that is common with other nodes is software configurable and can contain overlap portions with other nodes in the system. The Read Monitor contains a first-in first-out (FIFO) buffer that can queue up 64 memory write requests. This allows the intercomputer datalink to continue processing other nodes requests. The Read Monitor does not occupy space in the node's main card cage. It connects to the node on the back side of the bus directly behind the DPIMM. This board actually interfaces directly into port 1 of the Dual ported Memory module.

Intercomputer Datalink - Up to nine nodes can be connected on one intercomputer datalink, each daisy chained by twisted pair, differential ended cables. Minimum node intercomputer port hardware includes one (1) 2MB dual ported memory module, one (1) write monitor, one (1) read monitor, and datalink cables. The intercomputer hardware is capable of generating external interrupts in the event of hardware failures. Parity errors, buffer overflow conditions, non-present memory and error correction code (ecc) errors all generate an external interrupt which can be monitored and serviced by application software. This capability allows the user to determine the

integrity of the intercomputer datalink during realtime execution. If the datalink is failing, then an orderly software shutdown can occur.

A high reliability option minimizes single point of failure in a tightly coupled dual ported shared memory system. All datalink termination circuitry as well as the link arbitration circuitry is housed in a separate unit. This allows power off maintenance for any node in the system without disrupting or causing data loss on the intercomputer datalink. The datalink is a passive link so that off-line repair of the intercomputer port hardware can be accomplished without affecting the other nodes in the system.

COMPUTER SYSTEM HARDWARE

With the division of the simulation software into multiple tasks and the ability to execute them in a distributed environment it is necessary to change the hardware from the monolithic processor approach to a multiple processor approach and still consider the standard requirements of simulation, such as smaller footprint, increased performance, and lower cost.

To get from here to there you must first identify the areas that don't fit the new approach. Starting with a basic system, as shown in figure 3, we see that it takes at least ten printed circuit boards plus memory and input/output (I/O) devices to run the simplest of jobs. The traditional architecture of the Central Processing Unit (CPU) and the Floating Point Accelerator (FPA), though solid and well established, required too much chassis real estate. New requirements such as increased reliability, reduced cost, and more performance in less space have led to a new design that requires only two chassis slots, one for the CPU and one for the FPA.

The remaining boards, all I/O processors, also needed to be incorporated into a single PC board. This goal was less conceivable, but nevertheless obtainable if the application was Distributed Processing using discless nodes or if I/O performance requirements were not a major factor. Because Distributed Processing was the target application, redesign was a reasonable goal. Reducing a ten board basic system to a three board basic system allowed for redesigning and repackaging the system cabinet. Memory was also addressed by designing some of the features of two standard memory boards into a single board memory and by also adding a second port that could be used by the new shared memory system.

Now comes the question: How do you squeeze a three board set, a two board set, and five I/O boards into three single boards? Following is a brief overview of the philosophy and technique used to develop system hardware to build a tightly coupled distributed processing system.

The CPU board set answer seemed to lie in Custom CMOS Gate Array Technology. With the reduction ratio of 3:1, development decided to start with the 2 micron size arrays and attempted to squeeze the new design onto a ten layer PC board, but unknowns such as timing problems that required a timing circuit external to the arrays ate up large pieces of board real estate. Thus the newer 1.2 micron technology was used in some areas to meet the physical limitations of the ten layer PC board. In the pre-development stages, additional functionality was added to the project. The new CPU would now contain the capability to run virtual memory operating systems as well as real-time memory operating systems. The result of the new design was a single ten-layer PC board with a mixture of 2 micron and 1.2 micron Custom CMOS Gate Arrays that runs both real-time and virtual memory operating systems, and uses about one-third the power of its predecessor.

In parallel with the CPU design the task of combining five I/O processors into a single PC board was undertaken. Rather than risk a whole new system on a single new form of technology, it was decided that the Multi-Function Processor (MFP) should be designed from existing, time-tested, standard technology that engineering was more familiar with than the Custom Gate Array technology, which was used on the CPU. Preliminary layout indicated physical limitations would require the design to reside on one and one-half PC boards. However with the availability of connector pins on the backplane that was being used, the one-half size card could be a Device Interface (DI) card which would plug onto the rear of the backplane in the same slot location as the full size card: thus, the two cards would only use one backplane slot. Utilizing AMD29116 technology for the System Bus interface, Z80 and SCN2681 DUART technology for the asynchronous section, and Z80, NCR5386S and 8310 technology for the small computer system interface (S.C.S.I.) section resulted in a board set that is compatible with the System Bus, has two S.C.S.I. ports for disc and tape, seven asynchronous ports, one parallel printer port (Centronix or Data Products configurable), one console port, a real-time clock, an interval timer, and twelve external interrupts. All of this I/O was routed on ten layer P.C. boards that use only one backplane slot and consume only about half the power of the five processors that they replaced. The CMOS CPU and the Multi-Function Processor were designed in parallel and were started first because of their high degree of difficulty. After these projects were well under way, the redesign of the Floating Point Accelerator was started so

that its completion would coincide with the other projects.

The redesign of a two board Floating Point Accelerator to a single board Floating Point Accelerator utilized Surface Mount Technology (SMT) to simply reduce the size of the integrated circuits that were currently being used on the two board set. Use of SMT carried with it the prerequisite of a new P.C. board. So a new ten layer P.C. board with surface mounted chips was designed and built. The implementation of the SMT board was a one-for-one replacement of the previous floating point board set without attempting performance changes. With a one-for-one, gate-for-gate redesign the results would be more predictable and much simpler, plus the same arithmetic accuracy would be obtainable. The floating point redesign was completed in twelve months, which coincided with the completion of the CMOS CPU and the Multi-Function Processor. Now that the processors were completed it was time to add memory to the package to complete the board set.

To complete the hardware package a Dual Ported Integrated Memory Module (DPIMM) was added to the system. The DPIMM as its name implies, has two access ports: one port interfaces with the system bus and the other port interfaces with the new intercomputer memory system. Design features such as a four deep request latch on port 0 and two way on board interleaving give the Dual Ported Memory performance that is equal to or better than two standard memory modules. This memory board is discussed further in the "SHARED MEMORY HARDWARE" section.

By reducing the system board count from ten to three and one half and including a dual ported memory board we now had the basic complement of boards required for a stand-alone computer system. This opened the door for a redesign of the system chassis size from a fourteen slot version to an eight slot version. The eight slot chassis houses the three and one half boards plus a dual ported memory board and still leaves fifty percent of the chassis for future expansion. Mounting the eight slot chassis vertically in a 22" wide by 55" high cabinet allowed up to four chassis to be mounted in a single cabinet. The power supplies were also mounted on a vertical plate next to the chassis and both the chassis and power supplies were engineered so that they could be easily unbolted and slid out for repair or replacement.

The modularity of the chassis and power supplies made the system easily configurable from one to four chassis in a single cabinet or from one to eight chassis if two cabinets were bolted together. This Electro-Mechanical package lent itself well to the closely coupled distributed processing system that was also currently under development. With the front to rear air flow and four chassis per cabinet a host cabinet and two multi-chassis cabinets could be bolted together to form a nine processor distributed system in a footprint of five feet two inches wide by three feet two inches deep and four feet seven inches high (5'4" X 3'2" X 4'7"). See figure 4. The new hardware packaging shrinks the footprint by 361 Sq. Ft. by reducing the number of cabinets from nine to three. The smaller board complement reduces lifetime sparing requirements --especially when used in a tightly coupled distributed system where all of the basic boards are the same. The lower power requirements of the basic board set will increase the overall reliability of the system by reducing frequency of failures due to excessive heat build-up. The modular approach will serve two needs: 1) the need to easily expand a system, and 2) the need to easily maintain and service the system. All of these improvements will be beneficial in the Simulation Marketplace.

SOFTWARE CONTROL

The connection of multiple computer systems using the Tightly Coupled, Dual Ported Memory approach requires a central point of system control. Figure 5 depicts a Host and two nodes fully configured utilizing the dual ported memory approach. I/O controllers would be added based on simulation requirements. Node 0 (the Host) has control over all other remote nodes in this distributed computer system. Each node in the system is remotely bootstrap-loaded and receives all control commands from Node 0. The operating system executing in each node must be the same revision level as the Host system and retains all the capabilities of the Host system. This requirement is being mandated by simulation Request for Proposals (RFP's). The remote nodes do not require any peripheral equipment to load and execute the operating system, application software, or diagnostics. Peripheral equipment is supported on any node in the system should the specific simulation task require it. The central point of control does away with a requirement for a console crt and a hard disc drive on each node in the system just to support the bootstrap process.

Operating system and simulation code loading takes place via the RS-232 Control Link and the Tightly Coupled Dual Ported Memory Link. All operating system features are available to each task in every node in the system. Each node supports and executes an independent copy of the operating system; therefore, all application code executing in a node is fully supported at the system service level. Full operating system support eliminates the need and overhead of intercomputer operating system message

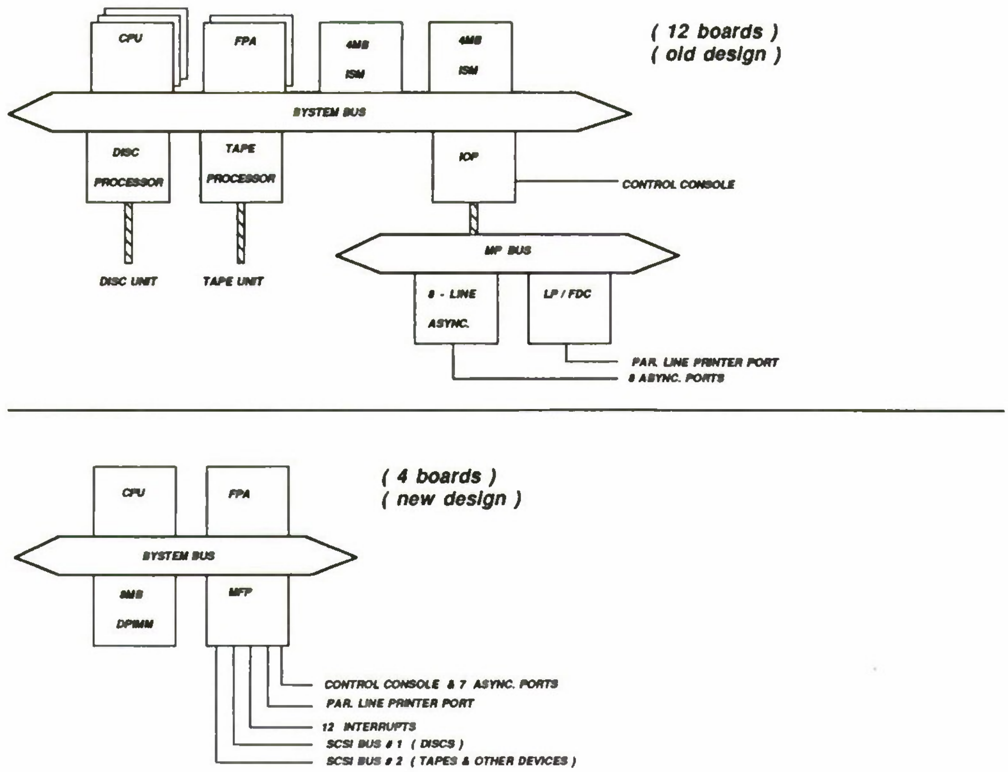


FIGURE 3

HOST & EIGHT NODES

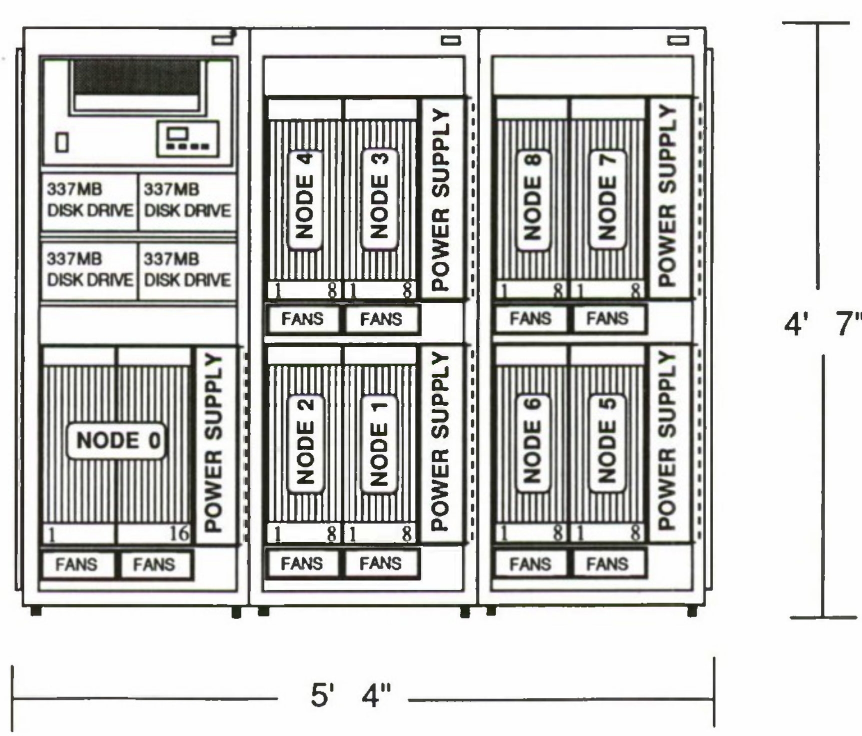


FIGURE 4

HOST / NODE DISTRIBUTED SYSTEM

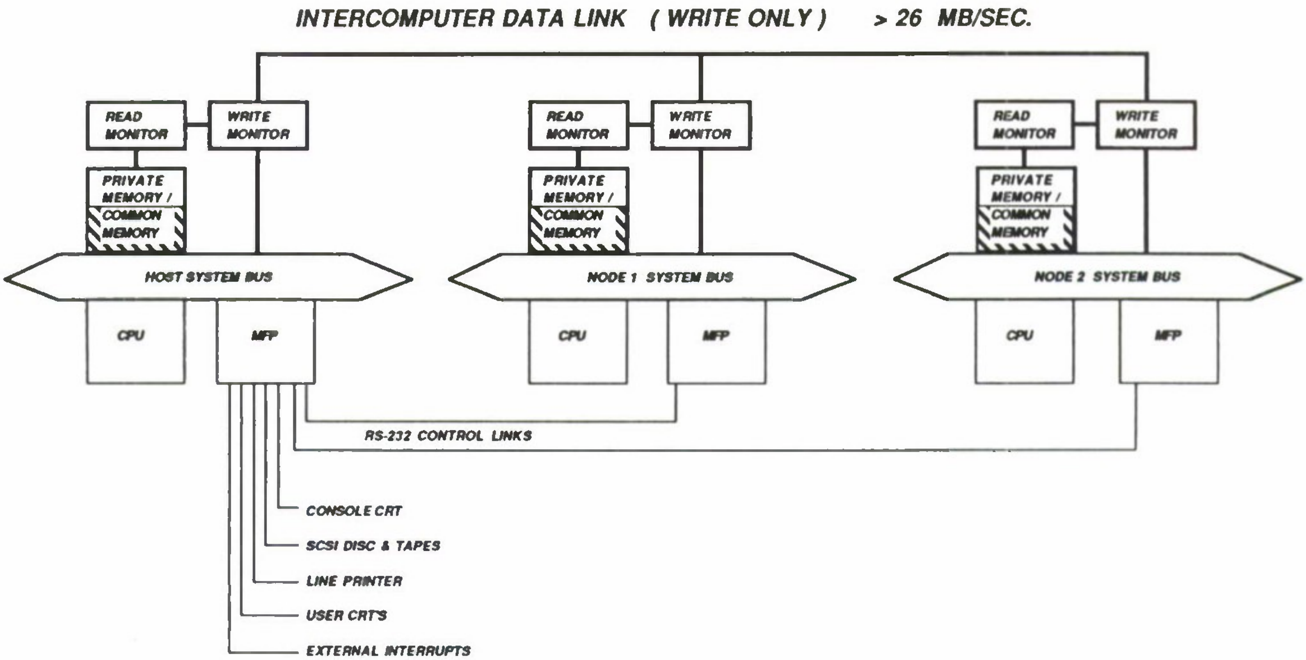


FIGURE 5

services over the Tightly Coupled Dual Ported Memory Link. The operating system that the application code executes under in real time must have the same support as the operating system that the application code was developed and debugged under. A full complement of hardware interfaces is supported by the operating system executing in each node. Each node in the system has the ability to perform all I/O independent of other nodes in the system. Application code has direct access to any configured hardware interface; therefore, the I/O load can be spread among all nodes in the system. Any I/O interface can have direct memory access into the area of memory being shared by all processors in the system thereby, providing all of the simulation tasks with access to the input buffer.

The software system gives the user the ability to preselect and build a remote nodes operating system and task load on the Host system. Once a remote node is loaded, the operating system can activate all simulation tasks at system start up. Memory disc support allows for high speed task activations and high speed disc I/O on a limited basis. A memory disc is defined to an operating system by configuring a portion of main memory to be formatted as a disc device. There are no rotational or head seek latencies associated with a ram memory disc. The memory disc is limited only by the amount of physical memory in a node. All access and use of the memory disc is transparent to the simulation code and the operating system sees and treats it like any other disc defined to the system. If the simulation requires hard discs on one or more remote nodes then the ability to bootstrap the node from the hard disc instead of the in-memory disc is also supported. In this case the memory disc would not be transferred to the node during the bootstrap process. Main memory that would normally be allocated by the memory disc would be available for other uses in that node.

A suite of software utilities is necessary for the central control of a Tightly Coupled Distributed Computer System. All control, downloading and monitoring of the entire system is from a single designated point -- the Host.

REMOTE BOOTSTRAP UTILITY

The Remote Bootstrap Utility is capable of bootstrapping from 1 to 8 nodes from the Host system (node 0). This utility is controlled by a master system definition file. This file defines the node configurations of the entire system. The file contains information defining each node's configuration to include: Node Operating System Image, Node Memory Disc Image, Control Link Address, Memory Disc Device Address, etc. The boot utility executes automatically when the Host system is initialized but can be inhibited from executing by setting a system flag before the Host system is activated. This feature prevents other active nodes in the system from being re-activated when the Host system is restarted. The bootstrap utility also has the ability to bootstrap

all nodes or selective groups of nodes in the system. Selective groups of nodes can be bootstrapped by using an alternate control file with the remote bootstrap utility. Selectively bootstrapping individual nodes does not effect the operation of other nodes in the system. The minimum hardware required to remotely bootstrap a node is a CPU, MFP (Multi Function Processor), a dual ported memory module and intercomputer datalink interface boards. Each remote node's operating system image is built, tested and saved on the Host system, using standard operating system utilities. All operating system initialization tasks as well as user application tasks are built into memory disc images for each target node in the system. A standard set of utilities is used to build and save the memory disc images into standard disc files on the Host system.

ERROR/MESSAGE MONITORING UTILITY

The Error/Message Monitoring Utility is automatically activated on the Host system by the Remote Bootstrap Utility after the first node in the system is initialized. The monitoring utility executes only on the Host system and continually monitors the console port of each active node for operating system and user generated console messages. Any system error or user generated message sent to a node's console port will be captured by the monitor and is retyped to the Host console crt along with the node I.D. of the message originator. In the event of Host system failure and restart the utility can be re-activated manually to continue monitoring all other active nodes on the system. The utility can detect Control Link I/O errors. If an RS-232 Control Link error occurs, then the node is marked offline to all utilities executing on the Host system. If a remote node halt condition is detected, then the monitor prints the appropriate halt message on the Host console crt and marks that node offline. To reestablish communications over the Control Link or to reactivate a node the Remote Bootstrap Utility must be executed.

FILE COPY UTILITY

The file copy utility executes on the Host system and allows disc file transfers from the Host to any configured and active node or from any node back to the Host file system. Disc files being transferred can reside in any volume and directory on the Host or nodes file system. The copy utility signals the target node via the RS-232 Control Link and transfers the disc files over the intercomputer datalink thru a static communications partition. The utility executes interactively or in command line mode. Interactively it is menu driven and prompts the user for all necessary input to complete the file transfer. In command file mode all necessary input is passed to the utility as parameters on the command line. If a parameter is incorrect, the utility will switch into interactive mode and begin prompting the user for the necessary input to complete the transfer. Simulation tasks or data files can be transferred or retrieved from the nodes by using this file copy utility.

REMOTE LOGIN UTILITY

The Remote Login Utility executes on the Host system and prompts the user initially for target node I.D.. It then establishes communications with the target node via the RS-232 Control Link. From any configured terminal on the Host system the user can remotely login into any active remote node in the system. The user can execute commands to determine the status of the node, directly execute file system commands as well as other system level commands.

Using the remote login utility there are two methods of debugging user written handlers that reside within the operating system. With the operating system debugger the user can set break points and stop execution of the operating system at any point. When the operating system halts at the set break point the user can begin displaying registers, changing memory, displaying queues to determine if his code is operating correctly. Additionally, the user can directly enter into the panel mode on the remote node to Halt the node and set instruction stops, write stops or read stops and then release the system to continue to execute normally. When the node encounters one of the set stop points, it halts and allows the user to single step from that point.

The Remote Login Utility has the ability to operate in batch mode so that a predetermined set of commands can be built and executed from a single command file on the Host system. With one command sequence the user can transfer an executable task to a node and get it activated by using the Remote Copy Utility and the Remote Login Utility.

REMOTE STATUS UTILITY

The Remote Status Utility is initially menu driven and executes on the Host system. It uses only the RS-232 Control Link for communications with the target node. It provides target node operating system status and task execution status at a preselected snapshot update rate. All status is displayed at the user terminal on the Host system. Any target node errors or user generated console messages that occur during the status update are captured and immediately redisplayed on the Host console crt. The remote status utility provides operating system status to include: % CPU availability, % IPU availability, % memory availability, O.S. image name, Time of day, etc. The task execution status returned includes: Taskname, Size, CPU & IPU accumulated time, state, etc. The status information displayed by the utility gives the user

detailed information about the condition and current loading of the system during realtime execution. It can also be useful in determining if a particular task is behaving correctly in the system or if the task is in an abnormal queue or if it's using an excessive amount of system resources.

REMOTE COMMUNICATIONS UTILITY

The Remote Communications Utility resides in memory of each active node in the system. It is automatically activated by the remote node's operating system at bootstrap time but, it can be optionally disabled. If disabled, the remote node status and remote file copy features are not available. It communicates with Host system utility programs using the RS-232 Control Link as well as the Tightly Coupled, Dual Ported Memory Link. It interprets opcodes it receives from the remote status and the remote copy utilities and interfaces with the remote node's operating system and file system.

DIAGNOSTIC SOFTWARE

All diagnostics including a diagnostic executive supports a multi-CPU tightly coupled environment. The diagnostics package down loads its executive as well as specific diagnostic programs into a target node. All input from the Host system is directed to the selected target node and all diagnostic output is redirected back to the Host system. The only peripheral devices required on a node are those required by the simulation tasks. No additional peripheral equipment is required to execute the diagnostics on any remote node in the system. All communications to and from the nodes is thru the RS-232 Control Link and the Tightly Coupled, Dual Ported Memory Link. Diagnostics can be executed on any standard supported hardware product that is properly configured on a node. Once a failure is detected, and the repair has taken place, then all that remains is to reinitialize the affected node using the remote bootstrap utility. At this point application software can be restarted and it can reestablish communications with other nodes in the system over the Tightly Coupled, Dual Ported Memory Link.

CONCLUSION

In the previous sections, the authors have shown the use of new methods and technologies for tightly coupling multiple computer systems for flight simulators. By using gate array and surface mount technologies the computer hardware system has been reduced to just a few boards. This has allowed smaller system packaging and will drastically reduce the life cycle costs of the computer system. The conventional shared memory hardware has been replaced with a high-speed dual ported common memory system. Central software control of this tightly coupled multiple computer system gives the simulator manufacturer the flexability required to configure todays simulator systems.

The solutions arrived at have made it possible to attain the high system fidelity required for flight simulation using a tightly coupled, distributed system.

ABOUT THE AUTHORS

Mr. Bocskor is currently product manager for the MPX-32™ real time operating system and the Multiprocessor Reflective Memory Software System, at Gould Inc. He holds a degree in computer science and has 15 years of hardware and software experience related to commercial and military flight simulators. Since joining Gould in 1981, he has served as a software consultant to a major simulator manufacturer and was involved with the design and development of the control software for the Tightly Coupled, Disributed Processing System.

Mr. Cichon is presently the hardware product manager for the CONCEPT/32™ product line, at Gould Inc. He has fifteen years experience in analog electronics and ten years experience in digital electronics. Twenty three years were spent in various technical and managerial functions in a customer services environment, and the last two years were spent as a hardware product manager, with the current responsibility of developing the system packaging and configurations for the Distributed Processing product.

THE DEVELOPMENT OF A REAL-TIME ADA EQUIPMENT SIMULATION

Scott L. Waldron
AAI Corporation
P.O. Box 126
Hunt Valley, MD 21030-0126

ABSTRACT

This paper will describe the software development effort that was made in the development of a receiver simulation using bare-machine Ada®. First a description of the host system will be given. After this concept is presented, the model selection and specification will be discussed. A brief explanation of the tools and methodologies (i.e. Ada compilers, bare-machine Ada, object-oriented design, DOD-STD-2167 waterfall model, simulation approach, ADADL® design tool) will then be given. The software design phases will be presented next, which include preliminary design, detail design, code and unit test, and hardware/ software integration. Finally, a later addition to the model followed by various techniques developed will be outlined. Assuming the reader may someday be involved in a similar endeavor, it is hoped that this start-to-finish style approach of presenting the software development of a real-time receiver simulation will be exceedingly beneficial.

INTRODUCTION/BACKGROUND

On a recent contract for the Electronic Systems Division of the Air Force Systems Command, there was a requirement to prototype a training workstation. This effort required the assembly of a training workstation and the design and development of a simulation package for the training workstation. The simulation package required real-time simulation software, hardware panels, and computer-based training (CBT) courseware.

The design had constraints imposed by both the customer and the design team. The customer required that the prototype training workstation match the design of the system as presented at the System Requirements Review (SRR). For the simulation package, the following features of the training system needed to be addressed:

- Interface with the training workstation
- Demonstration of digital audio capabilities
- Integration of fully-functional simulation panels

The design team added the requirement that the software be developed in accordance with the project software development plan.

Training Workstation

Before proceeding with this discussion, it is necessary that the reader understand the concept of a training workstation. The training workstation provides all the basic hardware, software, and courseware necessary for basic computer-based training. As can be seen in Figure 1, the training workstation is powered by an IBM® PC/AT computer system. CBT occurs at both the alpha/ numeric and graphics CRT's. The graphics CRT is equipped with a touchscreen for CBT interaction. The video disk player is under the direct control of the CBT program for single-frame or real-time video presentation on the graphics CRT.

The training workstation software consists of two main components. The first is the CBT program. This program provides structured training utilizing workstation hardware. Lessons are stored on the workstation disk and can consist of virtually any type of material. The CBT program has the capability to interface with the real-time modeling software in order to extract information for student testing, student evaluation, and lesson control.

The second component is the training workstation real-time executive. This program runs as an interrupt routine which is scheduled by the system clock interrupt and provides the training workstation interface to a simulation package. The simulation package consists of simulated equipment panels controlled by a real-time simulation model running on a 68020 microcomputer. The simulation package connects to the training workstation through an eight-line parallel bus.

MODEL SPECIFICATION

The requirements for the prototype specified features and components which must be used, but not a specific system to be simulated. The first task of the design effort was to select a system to model. Rather than select a real-world system, a system for prototyping was designed. The reasons behind a new design were twofold. First of all, a new design would emphasize the features of the workstation. Secondly, a new design would not lend itself to differences of opinion over the fidelity of a simulation model.

Because of the requirement to demonstrate digital audio capabilities, the obvious choice for the simulation model was an audio receiver. The receiver was designed to have two dedicated audio channels which could operate independently. A panoramic display was added to allow viewing of the entire frequency spectrum. A warmup cycle, backup antenna system, and built-in-test logic were to be modelled. These features would demonstrate all the customer requirements and provide courseware developers with a good basis for development of an integrated CBT lesson.

After the model was selected, a detail specification document was developed. This specification included the panel layout and man-machine interface. The document was distributed to the software, hardware, and courseware developers, and the design and development efforts for the prototype proceeded in parallel.

® Ada is a registered trademark of the U.S. Government (Ada Joint Program Office)

® ADADL is a registered trademark of Software Systems Design, Inc.

® IBM is a registered trademark of International Business Machines Corp.

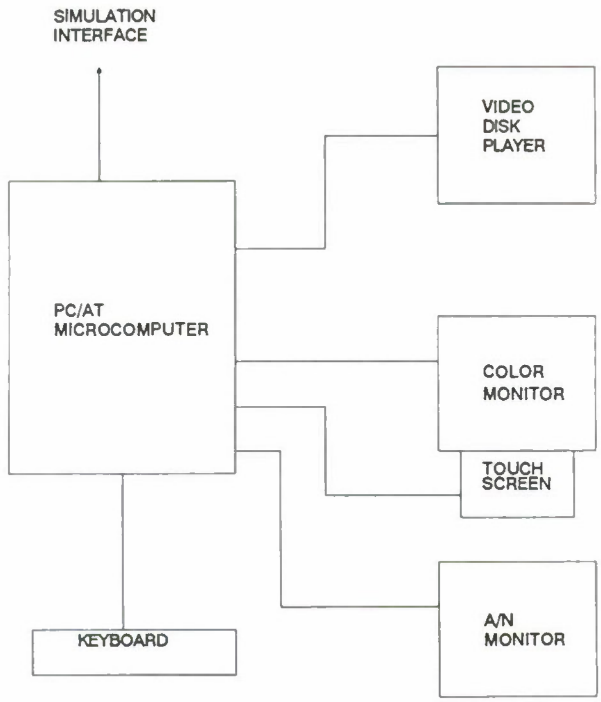


Figure 1. Training Workstation Block Diagram

TOOLS AND METHODOLOGY

The software development plan contained many new design requirements. The language required was Ada, and the design methodology was object-oriented design. The life-cycle model followed DOD-STD-2167's waterfall model. The software development was carried out on four different computer systems including a Gould PowerNode 9080™ 32-bit minicomputer, an IBM PC/AT, a Sun 3/160™ microcomputer system, and a Motorola MVME-68020 single board microcomputer. How these systems fit in to the life cycle stages will be explained later. A brief explanation of the tools and methodologies employed follows:

Ada Language

There was not a contract requirement for the prototype to be developed in Ada, however, AAI chose Ada over the other candidate, Fortran. The Ada compilers, debuggers, and support tools available were judged to be far superior. Ada was also felt to provide a better language for modular program development. In-house development studies had shown that the productivity rates for programmers using Ada were much higher. Internal training classes for Ada software design and development provided trained staff.

Ada Compilers

Four different Ada compilers were used for software development. These were the Telesoft Ada compiler for the Gould PowerNode 9080, the Alsys compiler for the PC/AT, the Verdix self-hosted Ada compiler for the Sun 3/160, and the Verdix cross-development compiler for the 68020.

Bare-machine Ada

Bare-machine Ada is an implementation of Ada where there is no operating system on the target machine - the Ada program provides all the necessary functions. The specific implementation of the Ada run-time environment must be tailored to the requirements of the target computer. The Verdix compiler achieves this goal with a configurable run-time kernel. The user is provided with the source code to an Ada package which can be modified for a specific installation. It contains machine-dependent attributes such as run-time stack location. The real-time clock interface is defined here. The user would insert the code necessary to start, read, and process interrupts for the specific clock circuit of the target processor. An example of how to do this is given later.

Object-oriented Design

Object-oriented design partitions the simulation task into items and operators. The items represent the data in the system, and the operators represent manipulations to the data. For example, switches on the panel can be represented as an item, and operators on the switches would be "turn on" and "turn off". Ada represents the items as packages and data types and the operators as procedures and functions. An executive program ties the control flow of the simulation together. This design approach is easily implemented in a language such as Ada.

Waterfall Model

Software development followed DOD-STD-2167's waterfall model. The software development effort consisted of preliminary design, detail design, code and unit test, and hardware/software integration. This life cycle model followed the development standard currently used by AAI.

Simulation Approach

The simulation model is a self-contained Ada program controlled from a cyclic real-time executive. The cyclic executive is just a subprogram scheduler running in an infinite loop. It provides an interface to synchronize the model to actual time.

The execution rate for the model was based upon certain elements of the equipment being simulated. Specific audio/visual cues had fixed rates (such as a 2 Hertz blinking lamp). Hand-to-eye coordination had to be considered (such as the tuning of a potentiometer which drives a meter). The design of the cyclic real-time executive allowed the rate to be changed dynamically so different rates could be tried. The rate chosen for the simulation model was sixteen frames per second.

ADADL Design Tool

The preliminary and detail design phases made use of a programming design language (PDL) as required by the software development plan. ADADL (Ada-based Documentation and Design Language) is a design tool which provided an Ada-based PDL, PDL compiler, and report generator. The application of the ADADL tool will be discussed later.

™ PowerNode 9080 is a trademark of Gould Inc., Computer Systems Division

™ Sun 3/160 is a registered trademark of Sun Microsystems, Inc.

SOFTWARE DEVELOPMENT PHASES

Each phase of the software development process provided a solid and verifiable product. The product was passed on to the next phase where additional design and/or code was added. This method provided immediate feedback on design quality and cohesiveness during each development phase. Not only was this useful from a software quality standpoint, but it provided immense programmer gratification as well. A discussion of each phase follows:

Preliminary Design

The top-level structure for the simulation model was developed during preliminary design. The objects of the system were isolated and represented as Ada data types. The operators were defined as Ada procedures and functions. Each object and its operators were grouped into an Ada package. The main control flow was designed using PDL. All of the package interfaces were defined using Ada "with" and "use" clauses.

This method of design yielded a state model for the receiver system composed of many smaller state models for each of the system's objects. As shown in Figure 2, the packages represented yield a design that is very easily related to the real-world equipment.

The development of the preliminary design was accomplished on a Gould PowerNode 9080 running the UNIX™ operating system. The design information referenced above was stored as Ada source files. The source files were compiled under the Telesoft Ada compiler which validated the package-to-package and package-to-subprogram interfaces and the Ada syntax. The ADADL tool provided checking for the PDL syntax and also checked PDL-package interfaces. The effort expended during preliminary design was approximately 30% of the total.

Detail Design

Detail design filled in the control flow and algorithms of the procedures and functions defined during preliminary design. This was accomplished through the use of PDL. ADADL was again used to compile the PDL. The PDL-to-package interfaces were checked. When problems with the package-to-package interfaces were discovered, the Telesoft Ada compiler was used to recompile the entire design.

This phase of development tied in very closely to preliminary design. Even though the PDL was added to the design during this stage, most of the control flow and algorithms were known at the completion of preliminary design. The designer involved in the preliminary design stage is best suited to complete the detail design. The product of detail design could be turned over to a programmer for code and unit test. This phase of the development cycle took about 20% of the expended effort.

Code and Unit Test

The code was written from the PDL and compiled on the Gould system using the Telesoft compiler. A hardware emulator had to be developed for software testing to map the physical hardware elements (lamps and switches) into logical quantities which could be monitored and changed from a CRT terminal. Testing of the software took place on the Gould using the hardware emulator. The code and unit test took about 20% of the expended effort.

Hardware/Software Integration

The software was ready for hardware/software integration (HSI) before the Ada cross-development system was available on the Sun. The panel hardware was ready, however, so the code was moved to the PC/AT. The code was recompiled using the Alsys Ada compiler and executed on the PC/AT with no changes

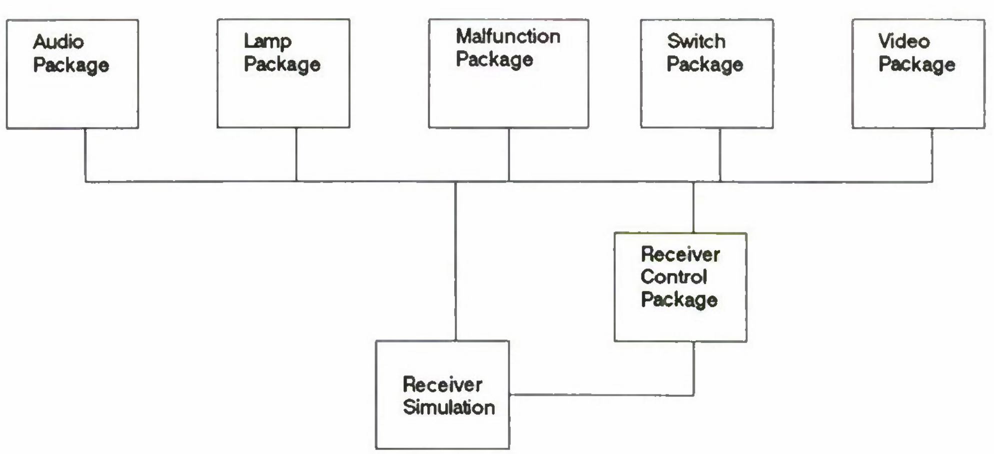


Figure 2. Simulation Model Structure

™ UNIX is a trademark of AT&T Bell Laboratories

required. The hardware emulator was replaced with an I/O driver which would map the model states into their appropriate hardware addresses using the eight bit parallel interface. The panel hardware was completely checked out on the PC/AT using the software model.

The Sun system was fitted with the Verdix self-hosted compiler before the cross-development compiler was available. The version of the model using the hardware emulator was compiled on the Sun using the self-hosted compiler and required only one small coding change. After successful compilation, the program executed as expected the first time.

When the Verdix cross-development Ada compiler was available, the model was ready to run on the target 68020. The I/O driver used for panel check out on the PC/AT was modified to interface with the memory-mapped I/O ports directly. The code was then compiled, downloaded, and checked out on the target computer.

The HSI showed quite a travel history. It was compiled under four different Ada compilers by three different vendors. It was executed on four different computer systems with three different design architectures. All of this was done with only one minor coding change. This truly is a testament to the portability of Ada code.

AN ADDITION TO THE MODEL

When the software development was complete and everyone on the design team had a chance to breathe, management decreed that another equipment function must be added to the simulation. A second panel was designed which added a maintenance function to the simulation. The new function required the development of another hardware panel and additional simulation logic. It interfaced to the original model only through the use of a common power switch. The details of this additional simulation are not important, but the effect on the existing model is important. The new simulation software was composed of the same types of operators and operations as the existing model. Very little change was required to add the new software to the object-oriented structure of the existing model. The object-oriented approach had yielded an easily modifiable design.

TECHNIQUES DEVELOPED

The configuration of the Verdix Ada run-time kernel required the addition of a clock interrupt handler. The interrupt handler that was developed was interesting due to the fact that it was written entirely in the Ada language. The clock for the 68020 is contained on a single chip. Register values are set to control the clock and read to determine elapsed time. The clock chip was modelled as an Ada record, with each element of the record being a clock chip register. The clock chip's address is known, so the use of an access type set up to point to the known address yields an Ada variable representing the chip itself. This variable can be used by an Ada program just like any other variable, with the hardware register being accessed rather than a memory location. Reference Figure 3 (shown on the following page) for a sample of this method.

A second technique developed for run-time configuration involved the establishment of the serial I/O channel to interface with the digital audio system. The run-time system buffered all serial input and output through two routines. The 68020 board contained a monitor chip which had a variety of supervisor calls available through a trap vector. The serial channel was established by modifying the run-time system's I/O routines to utilize the monitor trap vector which provided serial I/O. The modification was accomplished using the in-line assembly language provided by an Ada pragma.

CONCLUSION

This paper presented the software development of a real-time receiver simulation using Ada. The development involved tools and techniques which have not been commonly applied to the simulation field. The Ada language provided software transportability of source code between different computers using different operating systems. This transportability has not been observed at AAI with any other language. Bare-machine Ada provided a sound base for running the simulation model. The run-time system was easily adapted to the specific target microcomputer. Object-oriented design techniques proved to be a natural design methodology. The design partitioned the software into real-world entities which proved easily modifiable. Hopefully, the tools and techniques described may benefit the reader in similar endeavors.

ABOUT THE AUTHOR

Scott L. Waldron is a Design Analyst with the Training and Simulation Division of AAI Corporation. His principal responsibilities have been the design, development, and integration of real-time models into electronic combat training systems. He has received both a B.E.S. in Electrical Engineering and a M.E.S. in Computer Science from The Johns Hopkins University in Maryland. Recently he has been involved with in-house Ada development and training.

```
With Unchecked_Conversion;

Package Body Timer_Interface is

    -- Define data type to represent a 16 bit halfword

    type halfword is range 0..16#ffff#;
    for halfword'size use 16;

   -- Define the record structure which represents the chip

   type timer_chip is
   record
      gpip : halfword;  -- general purpose I/O register
      aer  : halfword;  -- active edge register
      ddr  : halfword;  -- data direction register
   end record;

   -- Declare a memory pointer to the timer_chip.

   type timer_chip_pointer_type is acess timer_chip;
   timer_chip_pointer : timer_chip_pointer_type;


   -- Define a routine which will establish timer chip address
   -- and utilize a register

   Procedure Setup_and_Use_Timer is

      --  Define a conversion routine to change an integer to
      --  a timer chip pointer, the size of the pointer is
      --  known to be an integer

      Function set_timer_chip_pointer is new
         Unchecked_Conversion (integer, timer_chip_pointer_type);

   begin  -- Setup_and_Use_Timer

      -- Set the timer chip's memory address to hex F80000

      timer_chip_pointer := set_timer_chip_pointer (16#f80000#);

      -- Once this is done, access the chip registers like a
      -- normal Ada record

      timer_chip_pointer.ddr := 16#01#;  -- write hex 01 to ddr

   end Setup_and_Use_Timer;

end Timer_Interface;
```

Figure 3: Sample Ada Code for Chip Access

LEARN TO FIGHT - LEARN TO TEACH: REQUIREMENTS FOR AIR COMBAT TRAINERS BASED ON FOUR YEARS' EXPERIENCE

A.G. Barnes, R.D. Armour
British Aerospace PLC, Warton, UK

ABSTRACT

The twin dome Air Combat Simulator at British Aerospace, Warton has been in regular use by the Royal Air Force to provide pilot training in Air to Air Combat. The training is given both at TWU (Tactical Weapon Unit) level, and are taught the basic skills and disciplines. OCU pilots are experienced squadron pilots who are taught the optimum deployment of their weapon system, and its capability against likely threats.

The simulator standard is described, with emphasis on the hardware requirements to provide high availability in rugged use. Features have evolved, particularly in he area of the instructor/operator station, to maximise the training benefit. These include rapid access to performance data, immediate selection of new configurations, efficient monitoring of performance, and instant replay.

The organisation of courses also contributes to training effectiveness. An environment is created to produce close instructor/student involvement. Students not participating in the actual combat benefit considerably by monitoring peer performance. The courses are short and intensive, without distraction.

Recommendations emerge relevant to the specification of training devices of this type. In particular, the cost aspects, and the technology trade-offs, are discussed.

INTRODUCTION

It is generally recognised that of all forms of flying, that required for air to air combat calls for the most developed skills. Pilots engaging in combat must have complete confidence in the performance capability of the aircraft they fly, and they must be able to use the performance right up to the limitations imposed by the airframe. They must have a similar understanding of the weapons which they will launch, and just what kind of a threat their opponents can offer. Added to this are the needs for rapid decision making in harsh physical conditions, and an appreciation of a complex three-dimensional scenario viewed from within, and the reasons emerge why top fighter pilots are a select few.

The success of the few does not depend on some mystical quality - all of them operate within a framework of rules of engagement; the tactics they use are open to analysis. Learning these rules and tactics in the air is both difficult and hazardous, and so the transfer of the learning task onto ground based simulators should be invaluable - provided of course that the simulator can be shown to do the task.

Such simulators have existed in both the USA and Europe for 10-15 years. In all cases they were developed in Industry or at a Government Research Centre, to assist in the design of new fighters, and to allow studies to be made of the operational aspects of new designs. In the course of such work, pilots have been quick to appreciate the contribution which such simulators could make to the air Combat Training task.

In view of this background, it is most surprising that Air Combat Training Simulators are not in widespread use by all Air Forces. The purpose of this paper is to relay some of the background experience in the United Kingdom from an Aircraft Industry standpoint. British Aerospace, has made extensive use of Air Combat Simulators for design and development, and emerging from this work has been a parallel activity in responding to the training needs from the Royal Air Force. As a result, we are able to offer views on what technology has to offer, and how it matches up to the customers needs.

AIR COMBAT SIMULATOR EVALUATIONS

Many of our research programmes over the past ten years have needed RAF front line pilots for the evaluations. The interest showed by these pilots led the Ministry of Defence (PE) to prepare a draft Air Staff Requirement for an air combat simulator for comment by Industry in 1979. The requirement called for the features available at Warton (and in other Air Combat Simulators) of projected images of sky, ground and target aircraft, inside a dome. Also included was a mode of operation in which the target aircraft could be computer controlled, and inter-active. Financial constraints on procurement had serious delaying effect on this initiative, although the Falklands crisis in 1982 did re-awake the UK in recognising that air-warfare has a part to play in military operations.

Consequently, the RAF asked for two evaluations to be made on the twin dome air combat simulator at Warton, to help in the preparation for the purchase of an Air Combat Training Simulator. The evaluations covered two aspects - the initial training of pilots in the basic skills of visual air combat, done by the RAF at their Tactical Weapons Units (TWU) - and the next stage of transferring these skills to the front lint - done by the RAF at their Operational Conversion Units (OCU).

Each of the trials, designed and conducted by RAF/MOD teams, consisted of using the simulator for a week. Instructors from these units advised the planners on the areas of training where benefits might be derived, and their predictions were tested by bringing half the students from courses about to begin, to get a direct measure of improvements.

Assessment Results

Both assessments came out strongly in favour of the procurement by the RAF of a twin dome ACS for training. In both of these assessments, the course was split, so that direct comparison could be made between students who had received, and those who had not received ACS training. The OCU report concluded:

"The results obtained at the end of the period and in correlation with those achieved on completion of the course, have shown the Twin Dome ACS to be a most valuable training aid, providing realistic ground simulation of air combat training. The OCU staff were able to demonstrate, supervise and monitor student performance in areas such as aircraft handling technique, energy management, air picture and tactical awareness. The results obtained in the ACS and in the air showed a positive correlation. The content of ACS profiles was identical to that of the air combat syllabus and ensured good continuity of training. Significantly, during the air work which followed on from the ACS work, no sorties were lost as a result of student inability and there was a noticeable improvement in their rate of progress."

The TWU report stated:

"Advantages of ACS Training. The advantages noted in student combat after flying the normal syllabus compared with non ACS training students were:

(a) Air Picture Awareness. The major advantage the students had was in air picture awareness. Their target prediction and lookout were better than average, enabling the early sorties to be learnt more effectively.

(b) Basic Combat Manoeuvre (BCM) Comprehension. The ACS gave students a better understanding of BCMs and their effects at an early stage of training. This understanding created a greater enthusiasm and interest in ACM, provoking discussion beyond that usually seen.

(c) Base Height Awareness. The introduction of a base height during the ACS training proved effective. Subsequently only the weakest student had any problem with base height during the flying phase.

(d) Student Predictability. It proved possible to predict fairly accurately student achievement in the flying phase by reference to his ACS performance. Some individual weaknesses were possible to detect, e.g. base height awareness and lack of aggression, so these could be worked upon before the airborne phase.

(e) Flying Hours. It is not felt that the ACS could replace any of the syllabus sorties but should certainly cut down the number of reflys of extra sorties required to bring students up to the required standard. It is noteworthy that no extra sorties were required by even the weakest student benefiting from the ACS training."

RAF SPECIFICATION AND TRAINING MODES

Description

The British Aerospace Air Combat Simulator, as supplied to the Royal Air Force, comprises:

- o Two Domes - each contains a cockpit and image projection equipment
- o Computing facilities
- o Visual generation system
- o Instructor's Console - for control and debriefing.

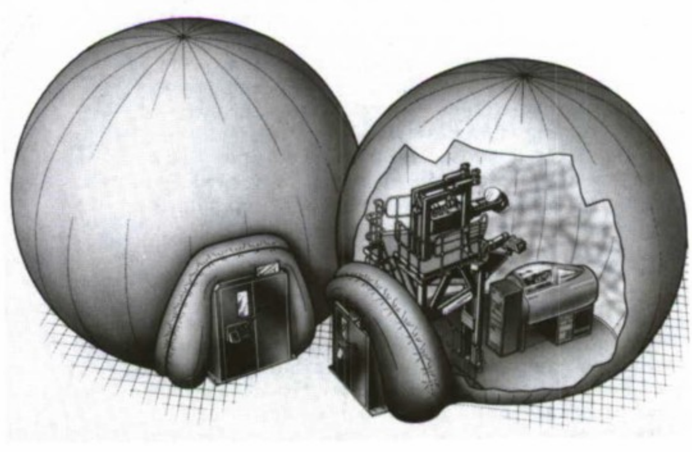


Figure 1 - Air Combat Simulator Domes

The cockpits are representative of the Tornado aircraft and are fitted with the instruments and controls necessary for air combat. The field of view from the cockpit is comparable with modern fighter aircraft. All the gantry/projector structure has been designed to be contained within a small, ±20° segment behind the seat. Cockpit noise, the effects of 'g' and buffet are simulated to a high degree of realism. Images of sky and ground are projected onto the inside of the dome, providing the horizon reference

which moves in relation to the manoeuvres.

The opponent is represented by the image of an aircraft projected onto the inside of the dome. The target image changes as a result of the manoeuvres carried out by the pilot and the opponent (which can be a second student, instructor or computer).

The performance of the simulated aircraft can be altered to match other aircraft types, including Harrier, Phantom, Hawk and threat aircraft. Target images can be changed quickly to give appropriate visual representation. Weapon performances can be also adjusted to simulate a variety of missiles. The host computer is the Gould SEL 32/9780.

Debriefing facilities allow the combat engagement to be viewed in a range of formats in real time and replayed as often as required. Important parameters can be display and recorded for further analysis:

Operating Modes

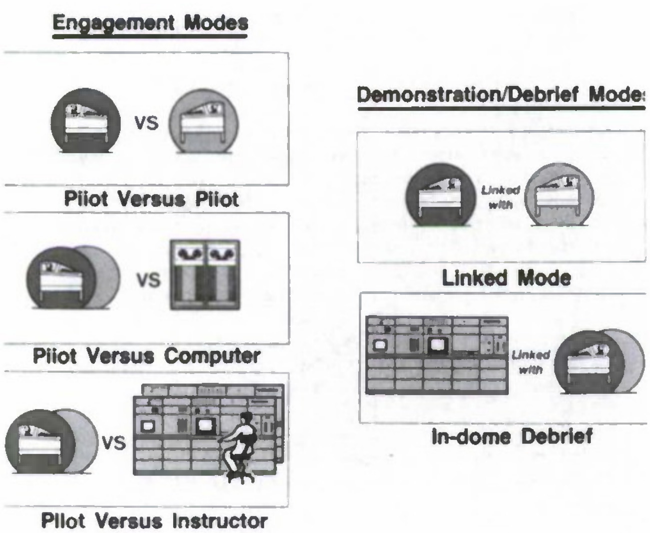


Figure 2 - Operating Modes

Pilot v. Pilot. The primary mode of operation involves a pilot in one dome manoeuvring against a target image which is controlled by a pilot training in the other dome. Thus pilot versus pilot combat is simulated.

Pilot v. Computer. Either pilot can fly against a computer-controlled opponent, which can be flown aggressively or non-aggressively, dependent on the task in hand, and used as a standard against which pilot performance can be readily and accurately assessed.

Pilot v. Instructor. This mode enables an instructor to fly the opponent aircraft using a miniature stick and throttles at the Instructor's Console.

Linked Mode. The domes can be operated in a Linked Mode where the scene in one dome is reproduced in the other. This enables an instructor to take control of the aircraft from one dome and demonstrate the desired manoeuvre to a pilot in the other dome.

In-dome Playback. The domes may be used during briefing or debriefing for playback of a previously recorded mission, providing a more realistic view of the combat engagement than can be displayed on the monitor at the Instructor's Console.

Instructor's Console

Figure 3 - Instructors Console

The Instructor's Console provides a means of ensuring the instructor has the optimum facilities for training purposes. The console offers the following features:

- o Selection of the sortie parameters:
  - Aircraft types
  - Weapons types
  - Combat starting positions
  - Fuel states
  - Computer pilot skill levels
- o Fully animated pre-briefing,
- o Control and communication,
- o Real-time monitoring of the combat and student performance,
- o Participation by an instructor, against a student in the dome,
- o Recording and playing back engagements, for debrief.

The Instructor's Console incorporates high resolution colour monitors. Engagements can be viewed in real-time or replayed for debriefing. A range of display modes can be selected, including

- o The Air Combat Manoeuvring Display. This display mode provides an external view of the combat. The view can be altered by zooming in or out, and by varying the viewing height and direction. There are two formats:
  - 3-D View. Shows an elevated view of the engagement, viewed from any position
  - Plan View. Presents a view of the combat from directly overhead.

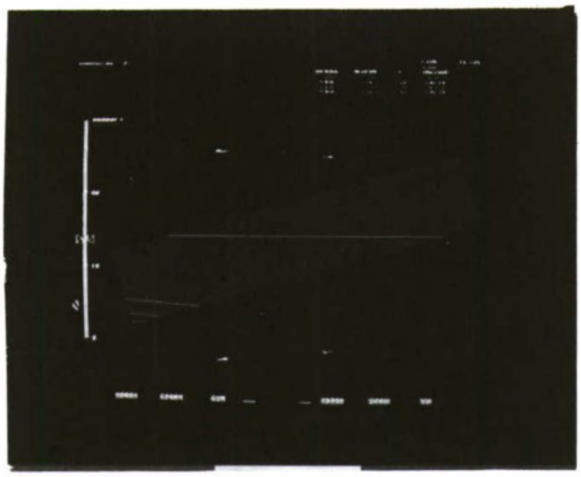


Figure 4 - Air Combat Manoeuvring Display

- o Inside-out Display. The pilot's view from either cockpit, including head-up display and status information. The field of view is variable.
- o Instrument Display. Relevant cockpit instruments can be displayed on demand.

EXPERIENCE

Following the evaluations described in section 2, the RAF saw that every student destined for combat flying could benefit from experience on the Warton ACS. Consequently, the simulator has been leased to the RAF on a regular one-week-in four basis since 1983.

The RAF have been sending 2 courses per week typically consisting of 10 students and 2 Instructors on each course. 86% of courses have been from the Hawk TWU's and the student is exposed to air to air combat in the simulator before the airborne phase of his course. A typical course consists of familiarising the student with the layout of the non-standard R & D cockpit and the cues of the ACS. After this brief period he will execute the basic combat manoeuvres against a pre-programmed and non-aggressive computer controlled target from various start positions. The Instructor will then go into the other dome, take control of the lead aircraft and start to execute mild defensive manoeuvres. The Instructor can immediately assess the students' response to a whole variety of geometrical situations and correct them if necessary.

The student, having watched all the other students' performing those manoeuvres would then progress to the next stage; that of engaging the aggressive inter active computer opponent BACTAC. The use of this opponent gives an immediate ranking of student performance and indicates if any necessary remedial action needs to be taken. They often finish off the course with a student knock-out competition. A small financial stake by each competitor adds to their competitive edge. During the course each student will have participated in about 30 5 min exercises and will have benefited greatly from observing the performance of his peers, viewing either from inside the dome, or at the Console.

The course develops a close Instructor/student relationship: they are off base for a week, with no distractions from the task of learning about Air Combat. Discussion does not stop at the end of the working day. The trepidation some of the students may have felt for the air combat phase of their course disappears, and at the start of the flying phase, Instructors must now watch for over-confidence.

One interesting point is that there has been no evidence of nausea, unlike some US experience, and each course is closely monitored by a questionnaire issued by the Institute of Aviation Medicine.

The other users of the simulator are either OCU's or pilots from operational squadrons, supplementing ACMI work.

The advantages over ACMI is that it is possible to study threat aircraft capabilities, and to prepare for weapon system developments before their introduction to the squadron.

MAN V. MAN COMPARISON WITH BACTAC

BACTAC is a computer programme developed by British Aerospace to replicate the tactics used by a pilot in close combat. It is used extensively, both for research work and in the pilot training courses which we regularly give to the Royal Air Force. The tactical rules it uses have evolved over several years of development in the nineteen seventies.

To engage the pilot, BACTAC continually re-assesses its view of the fight. It examines whether the piloted aircraft is:

ahead or behind,
pointing towards or away,
the range, and the range capability of the weapons.

From these decisions follow the choice of aggressive manoeuvres, defensive manoeuvres, or less extreme manoeuvres which include energy gain or conservation. Ground avoidance is another possible manoeuvre, and has priority over most other demands. The aggressive manoeuvres are sub-divided into regions of increasing threat to the opponent.

It is usual to justify the behaviour of programmes such as BACTAC by reference to pilot testimony. Controlled experiments to compare directly the success of a computer opponent with a pilot are rarely made (or rarely discussed). Four years ago, we conducted such an experiment. Six RAF squadron pilots flew a large number of close-combat engagements, against either BACTAC or each other. The aircraft types were the Northrop F5E and the McDonnell Douglas F4. Both aircraft were armed with rear-hemisphere IR missiles.

Scoring measurements and pilot comments were recorded, together with all parameters needed to reconstruct each fight. Table 1 shows some of the measurements.

Table 1

	Average No. of shots		Average IAS knots		Average g	
F5 man v man	0.11	0.56	256	265	3.0	3.1
F5 man v BACTAC	0.39	0.11	255	244	3.3	3.0

In the case of the F5 v F5 fights, a good validation of BACTAC's logic was obtained. The scatter in the number of shots was less than in the man v man case. The speeds and the g levels are similar. Pilot opinion confirmed that BACTAC was fighting in a similar manner.

COST/EFFECTIVENESS

What does it cost to train for Air Combat in the air? Costs are a sensitive topic, partly because they are useful for comparative purposes, and in such comparisons, the same basis for costs must be used. In a word, however, the answer is 'expensive'. Published information gives an indication of the order of costs. Training an RAF pilot to the point of joining an operational squadron costs around $5 million; only relevant in this discussion if a pilot is lost in a training accident, and a replacement necessary. Similarly, aircraft attrition in combat training is expensive; $10 million per aircraft as a minimum.

Hopefully, accidents are infrequent; the real cost is in the flying hours. Hourly flying costs clearly depend on aircraft type, and whether these costs should include all overheads, including aircraft procurement and spares. A typical range of published figure goes from around $5,000 per hour for an advanced trainer such as the Hawk, to $10,000 for front-line defence aircraft. For a simple one v one engagement, two aircraft are needed, weather and airspace must be suitable, other operational aspects add to the cost. Actual combat engagements in that hour depend on circumstance, like initial separation and control of air space; an average of three is realistic.

The ACMI is a good air combat training arena, with all the benefits of briefing, monitoring, and replay which have been described earlier. The RAF buys about 1000 half hour slots per year on the ACMI in Sardinia; each slot cost $4,500. The crews, aircraft, and ground support have to get to Sardinia. Add it all up, and air combat training is 'expensive'.

What does it cost to train for Air Combat on the ground? This is an easier question to answer. The RAF ACS described earlier today would sell for $7m; technology developments since its specification in the early eighties should allow a simulator with multi-combat, low-level combat capability, including missile fly-out simulation, for less than $15m. How does all this relate to the training cost per hour in the simulator? A week of training as described in section 4, might cost around $25,000, and depending on utilisation, $750 per hour emerges as an all-inclusive leasing cost to the customer.

With the above figures, it is easy to prove that of all ground training aids, Air Combat Simulators are the most cost/effective pilot training simulator on the market, including the well-established Commercial Airline Training Devices.

The unfortunate paradox is that most Air Forces see Air Combat Simulators as luxury items. In broad terms, they cost about the same as a full mission simulator, and they do not do the full mission (neither does the full mission simulator). The full mission simulator is essential - the ACS can be given less priority, because Air Combat Training has to be done in the air. In paraphrasing this customer view, the view sometimes held at air staff level must be added, that simulation of Air Combat still has some way to go. By delaying a procurement decision, a better product which is in the pipeline will emerge, and that is the one to buy. The fallacy of this argument is that the 'expensive' training tap is running right now, dollars are flowing, and some of that flow could be used more effectively, starting today.

CONCLUSION

The technology for Air Combat Simulation was developed for research and development, years ago. This technology has been applied to pilot training for Air Combat with conspicuous success. We have described the experience at British Aerospace in this area. The experience covers both the transfer of the design from a Research Simulator to a Training Simulator, and the operational aspects of using such a simulator to train pilots. The economics of this operation have also been presented.

Air Forces have been slow to recognise the monetary benefits which come from the use of simulators for Air Combat Training, but there is now every sign that the situation is changing. Adding support to this change of policy is the wider choice of training device which industry can offer. The basic ACS trainer has been there for several years. Today's technology not only offers these effective, low cost devices for teaching the basic skills, but also a wide choice of options, all cost-effective, to supplement the specialised training needed for tomorrow's fighter pilots.

ABOUT THE AUTHORS

Mr. Arthur Barnes is the Chief Flight Simulation Engineer of the Military Aircraft Division of British Aerospace. He is responsible for all aspects of simulation both for the companies Research and Development Simulators and Training Simulators and has over 30 years experience in the field.

Mr. Barnes was educated at Manchester University and is a Chartered Engineer and a Fellow of the Royal Aeronautical Society.

Mr. Ron Armour is the Group Leader responsible for all activities in the design and development of Training Simulators for the Military Aircraft Division of British Aerospace.

Educated at Preston Polytechnic, Mr. Armour is a Chartered Engineer and a Member of the Institute of Electronic and Radio Engineers. He joined the company in 1963.

THE DEVELOPMENT OF DESIGN GUIDELINES FOR MAINTENANCE TRAINING SIMULATOR INSTRUCTOR AND STUDENT STATIONS

Daniel J. Dwyer
Naval Training Systems Center
Human Factors Division
Orlando, Florida

Robert J. Carroll
Applied Science Associates
Butler, Pennsylvania

ABSTRACT

Trainer-critical features (e.g., performance monitoring, student recordkeeping, etc.) for maintenance training simulators (MTSs) are typically derived during the front-end analysis phase of the acquisition process. The critical features (i.e., functional capabilities) are then designed into the MTS instructor station (IS) or student station (SS) by incorporating these requirements in the procurement specification. Although many of these features are common to most MTSs, a lack of standardization in their implementation has led to vastly different operating formats despite the same instructional intent. This paper discusses the procedures and the results of a research effort to develop a tool for acquisition personnel and design engineers to ensure the standardization of critical IS and SS features during the design of the MTS. The procedures used during this research effort included (1) developing a classification scheme for categorizing the various types of MTSs, (2) developing a MTS attribute taxonomy to identify and categorize MTS features, (3) performing a commonality analysis to assess the degree of functional similarity of features across and within MTS categories, and (4) conducting a survey of instructors to determine users' perceptions of the effectiveness of the various features. The results of the survey indicated that instructors gave high (perceived effectiveness) ratings to 13 of the 17 features assessed. These results were relatively consistent across the different types of MTSs indicating that the features were a function of instructional requirements rather than peculiar to specific MTS types. The findings were then used to derive a set of design guidelines for developing maintenance training simulator instructor and student stations.

INTRODUCTION

An examination of maintenance training simulators (MTSs) in the Navy inventory reveals a variety of device configurations and types (e.g., 2-D panel simulators, general purpose or "generic" simulators, 3-D replica simulators, videodisc-based systems). Undoubtedly, the different types of MTSs are a function of both the different training requirements derived during the front-end analysis phase of device procurement, and the unique characteristics of the end-equipment which is being simulated. Additionally, it is evident that auxiliary components such as instructor stations (ISs) and student stations (SSs) associated with these training systems, also vary considerably across MTSs, both in their design and how their functions are utilized. In spite of this diversification, many functional capabilities are common to most MTSs. However, these capabilities often exist in very different formats despite the same instructional intent.

By providing a means for standardizing critical functional capabilities across MTSs, the Navy may be able to (1) reduce procurement costs by eliminating the need to design ISs and SSs each time a new system is procured, (2) ensure that training-critical features are given proper consideration for inclusion during the design process, (3) improve integrated logistics support (ILS) by promoting commonality across MTSs, and (4) improve user acceptance by facilitating transferability of user skills across simulators. Standardization has been suggested by several authors (Carroll et al,. 1984; Hritz and Purifoy, 1984; Nauta, 1985) and is advocated by Naval Training Systems Center Instruction 4120.3D (1984).

The objective of this research effort was to derive a set of design guidelines, based upon past research and the data gathered from a survey of maintenance instructors, to support the development of MTS instructor and student stations. The intent of the guidelines is to ensure that appropriate consideration is given to incorporating critical functional capabilities during design and to maximize the standardization of these capabilities across MTSs. Although space limitations preclude a detailed discussion of the specific implementation recommendations, they are addressed in Carroll et al. (in preparation), which provides a thorough discussion of the guidelines for the functional capabilities and presents them in the form of a prime item development specification. This paper describes the approach taken in the research effort, discusses the results obtained, and identifies and defines those functional capabilities deemed critical for training by maintenance training instructors.

TECHNICAL APPROACH AND FINDINGS

The development of design guidelines to promote standardization of training-critical features was based upon a systematic approach which covered a number of issues related to MTS acquisition. The approach taken, and the findings associated with each phase of the effort are discussed below.

Classification of MTS Types

While several different definitions of MTS appear throughout the literature, for the purpose of this paper, MTS refers to a class of maintenance training devices that represent actual equipment or systems via computer controlled simulation of equipment operation and responses to user input. They are necessarily driven via an auxiliary computer and are designed to duplicate the performance characteristics of operational (i.e., actual) equipment under normal and malfunction conditions.

Since it was possible that some functional capabilities may have been peculiar to specific types of MTSs, it was necessary to examine the functional capabilities in the context of simulator type. A review of the maintenance training literature revealed the lack of a standard classification scheme for categorizing the different MTS types in a commonly accepted format. Thus, the initial step in this research effort involved the development of a classification system for categorizing MTSs by type. First, existing taxonomies in the literature were reviewed in terms of the classification categories used, descriptions of MTSs that fit within these categories, and the training objectives, characteristics, and functional fidelity associated with each category. Next, the taxonomies were evaluated in terms of comprehensiveness, clarity, parsimony, and ease of use - factors which would promote application to the current effort. Finally, the reviewed taxonomies were synthesized, incorporating the strongest features of each such that the resulting classification system was composed of categories which were meaningful, non-redundant, and represented true discriminations between MTS types.

As a result of this analysis, four categories of MTSs emerged in the classification system: interactive video display simulators (IVDSs), panel simulators, model simulators, and stimulated actual equipment (SAE).

IVDSs include simulators which use computer-controlled videodisc images, computer-generated graphics, random-access slide systems, or any combinations of these formats. Typically, IVDSs are microcomputer controlled and consist of an interface device (keyboard/pad, touchscreen, mouse, etc.) and a video display unit for presenting images of the equipment the student is learning to maintain, and supporting information such as instructions, feedback, etc. An example of an IVDS is present in Figure 1.

Figure 1. Example of an Interactive Video Display Simulator.

Panel simulators (see Figure 2) are flat panels which contain simulated controls, test points, and displays. These components are configured in a manner which conveys their location and functional relationships in the actual operational equipment. Some controls, test points, and displays are functionally operative and are used for practicing hands-on maintenance tasks. Other components are merely photo-etched on the panel and are non-operative.

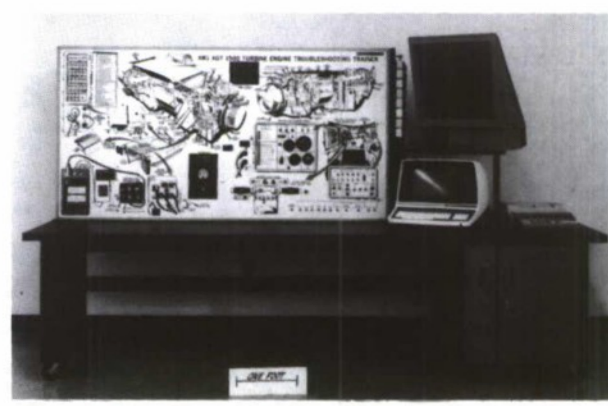


Figure 2. Example of a Panel Simulator.

Model simulators are 3-dimensional mockups or replicas of actual equipment. They may be full scale, under-scale, or enlarged representations. Typically, only those controls and displays essential to the tasks to be trained are functional; others are nonfunctional replicas or photo-etched. The functional components are used to support maintenance training via hands-on practice. An example of a model simulator is provide in Figure 3.

SAE refers to actual operational equipment which is directly stimulated by an auxiliary computer or some other input device (e.g., fault insertion device, signal generator). In the case of SAE, the actual equipment does not receive its input from normal sources, but rather from some external signal source, typically under computer control.

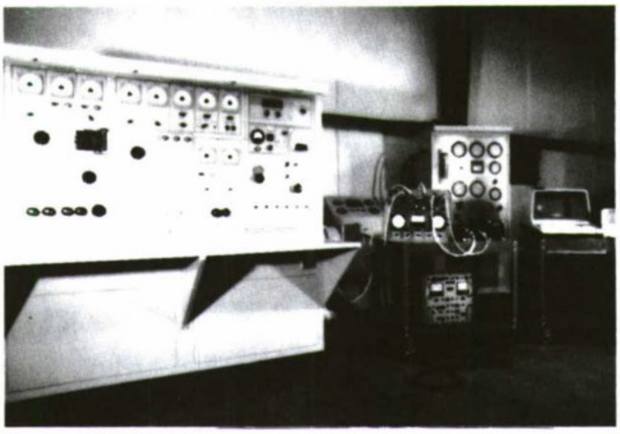


Figure 3. Example of a Model Simulator.

SAE is included in the classification system in order to provide a comprehensive taxonomy. SAE, which might be more accurately conceived of as a special case of Technical Training Equipment (TTE), is essentially off-the-shelf operational equipment which has been modified in some manner to enhance its training capacity. Because SAE is not truly a simulation system, the results discussed in this paper do not necessarily apply directly to SAE, but rather focus on IVDSs, panel simulators, and model simulators.

<u>Selection of MTSs for Study</u>

An initial list of 64 MTSs was identified for possible inclusion in this research effort. Several criteria were created which permited selection of a representative sample of MTSs from the initial candidate list. Each of the original 64 simulators was assessed against the criteria for incorporation in the study. The criteria used were:

(1) The device must be a dedicated maintenance simulator.

(2) The device must be used to train Navy and/or Marine Corps personnel.

(3) The device must have been used in a maintenance training course within the past 6 months.

(4) The device must be computer-driven for training purposes.

(5) The device must contribute to the goal of obtaining a representative sample of MTS types.

Those simulators which met all of the criteria were selected for study. The final sample consisted of 16 MTSs: 3 IVDSs, 7 panel simulators, and 6 model simulators.

<u>Development of an MTS Attribute Taxonomy</u>

In order to organize the functional capabilities around a conceptual framework, it was necessary to develop an MTS atttribute taxonomy. A review of the literature did not reveal any existing attribute taxonomies. However, various MTS attributes and features evident throughout the literature, were extracted and analyzed. Based upon this examination and the authors' experience with MTSs, a three-tiered taxonomic hierarchy was generated. The taxonomy consisted of four global attributes at the "top" level of the hierarchy, several features associated with each attribute on the "middle" tier, and multiple elements which represented subcomponents of each feature at the "bottom" level. This relationship is depicted below.

Attribute

Feature Feature Feature

Element Element Element

Four MTS attributes emerged from the analysis: (1) Information/Training Management, (2) Instructional Features, (3) Human Factors Layout and Design, and (4) Computer System Characteristics. The first, Information/Training Management, refers to a capability that provides the instructor with the ability to perform training administration functions via the simulator's computer system. This attribute is composed of features such as system initialization, performance monitoring, performance measurement, system monitoring, report generation, student recordkeeping, student tutoring, training exercise selection, and training exercise creation/modification. Each of these nine features, in turn, is composed of several elements. For example, report generation is composed of (1) summary reports and (2) statistical profile; performance monitoring consists of (1) sensing and (2) recording.

The second attribute, Instructional Features, refers to mechanisms of the simulator and the associated software which enable the instructor to control critical aspects of the learning environment. Features associated with this attribute include student sign-in capability, malfunction insertion, freeze capability, augmented feedback, next activity control, cue enhancement, system parameter control, and training mode control. Again, each feature can be further subdivided into a number of elements.

Human Factors Layout and Design is the third taxonomy attribute. This refers to the design and layout of system components (both hardware and software) in order to effect an optimal user-system interaction. This attribute addresses those user-system interactions which are under software control and mediated through the simulator's input and output hardware. The features associated with this attribute are input/control devices, display devices, workstation layout and design, and user-system software interface.

Computer System Characteristics, the fourth attribute, addresses the hardware and software characteristics (configuration and function) of the MTS computer system and subsystems. The features of this attribute were derived from Hritz and Purifoy (1984) and include instructional systems software, computational subsystem hardware, computational subsystem software, and trainer support subsystems.

Commonality Analysis

In order to determine if certain MTS features were unique to a particular MTS type, a commonality analysis was performed. This phase of the research effort involved a determination of the frequency with which each of several features appeared in the MTSs studied. The determination was made via on-site administration of a survey questionnaire to instructors experienced with the MTSs used in the analysis.

Fifty-one instructors, distributed across the MTSs selected for study, completed surveys which were designed to ask which of the features were present on a given simulator. If the instructor indicated that a particular feature was present, he was then asked to indicate on two 7-point scales, the extent to which he believed that that feature contributed to training effectiveness and how frequently it was utilized. If the feature was not present, he was asked to indicate how desirable it would have been to incorporate it within the simulator. (The results of this "criticality" assessment are presented later). Only those 17 features associated with the first two attributes (Information/Training Management and Instructional Features) are addressed in this paper since they may be properly categorized as functional capabilities. Those features associated with the Human Factors Layout and Design attribute are not reported here since they cannot be categorized as functional capabilities per se, but rather represent design features such as input devices (e.g., keypads, switches), display devices (e.g., monitors, digital counters), workstation layout, and user-system interface. Additionally, the features associated with the Computer System Characteristics attribute were not addressed in the survey because it was believed that the instructors would not have the information necessary to give sufficient answers to items concerning aspects of computer system hardware and software.

The survey data which dealt with feature presence were extracted and arranged in a matrix in order to determine if feature presence exhibited any pattern either within or between MTS types. MTSs were grouped by type and presented along the horizontal axis. Features were presented along the vertical axis. A mark in a given cell of the matrix signified the presence of that feature in that simulator. Patterns of feature commonality were then examined by visually scanning the matrix. The commonality matrix is presented in Figure 4.

FUNCTIONAL CAPABILITIES	IVDS			PANEL							MODEL					
Initialization	■	■	■	■	■	■	■	■	■	■		■	■	■		■
Performance Monitoring	■	■	■	■	■		■	■		■	■	■	■	■		■
Performance Measurement	■	■	■		■	■	■		■		■	■	■		■	
System Monitoring	■			■	■			■		■	■	■	■	■		■
Report Generation	■	■	■		■			■		■			■			■
Student Recordkeeping		■	■										■			
Exercise Selection		■	■		■	■		■					■	■		■
Exercise Creation/Modification		■	■										■	■		
Malfunction Insertion	■	■	■	■	■	■		■	■	■	■			■	■	■
Freeze					■					■		■	■			■
Next Activity Control		■			■	■	■	■					■	■		■
System Parameter Control		■		■						■				■	■	
Training Mode Control		■			■						■					
Cue Enhancement	■	■		■						■			■	■		
Augmented Feedback	■		■	■	■	■	■	■						■		
Student Tutoring		■			■									■		
Student Sign-In		■		■	■									■		

Figure 4. Commonality of Functional Capabilities Across MTSs.

The results of the commonality analysis indicated that, in general, most of the 17 features (i.e., functional capabilities) were present in all MTS types. A relatively high level of feature commonality appeared both within and across MTS types. This pattern suggested that, in most cases, feature presence was independent of MTS type, and that these functional capabilities tend to cut across all MTSs, regardless of type. A few exceptions, however, should be noted. Student recordkeeping and training exercise creation/modification were virtually non-existent in the panel simulators; the freeze capability apparently did not exist in the IVDSs studied; and both training mode control and student tutoring each appeared in only three simulators (one IVDS, one panel, and one model).

Criticality Assessment

The survey data which assessed the criticality of each feature were analyzed and a criticality index was generated (i.e., a composite effectiveness - utilization - desirability score) for judging the importance of each feature. Instructor ratings were averaged for each feature and the average ratings were placed in one of three "criticality bands", indicating a low criticality rating (criticality index was less than or equal to 3.0), a neutral rating (index was between 3.0 and 5.0), or a high criticality rating (index was greater than or equal to 5.0) for that feature. The results are presented in Figure 5.

The results indicated that only one feature (system initialization) was given a low criticality rating. Three other features (student recordkeeping, freeze capability, and training mode control) were rated as neutral. The remaining 13 features were rated as high, suggesting a strong belief by instructors that these features contribute positively to training function.

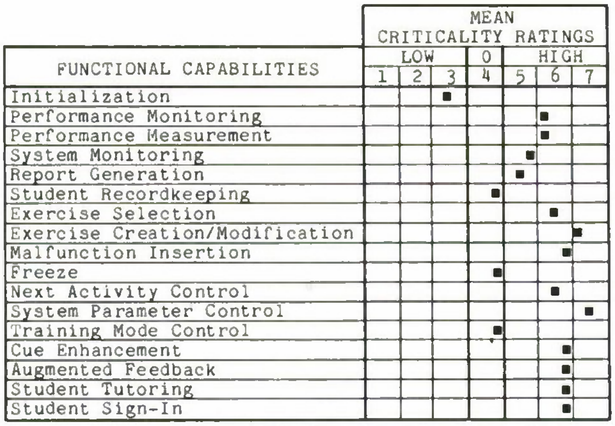


Figure 5. Mean Criticality Ratings for Functional Capabilities.

CRITICAL FUNCTIONAL CAPABILITIES: IDENTIFICATION AND DEFINITION

The 13 functional capabilities rated as highly critical by maintenance training instructors are presented below. These functional capabilities are briefly addressed here; a more thorough discussion is provided in Carroll et al. (in preparation).

Performance Monitoring refers to a computer system capability that automatically monitors (sense and records) student responses on a given training exercise. The advantage of this capability is that it allows responses to be recorded and later used to review specific areas of difficulty encountered by the student. This feature is a necessary prerequisite for both the performance measurement and report generation capabilities. The feature should be capable of being enabled/disabled by the instructor in order to conserve computer processing requirements when the feature is not needed.

Performance Measurement is a device capability that utilizes the simulator's computer system to compare student training performance to some pre-established criterion measure, assign a score, and store the results. Ideally, the instructor should have the control to adjust the criteria values against which student performance is judged. This capability allows the instructor to make qualitative judgements about a student's skill level.

System Monitoring is a capacity which provides the instructor with information about the control positions and display indications on the simulator during a training exercise. This allows the instructor to monitor student performance on-line, while the student is engaged in a practice scenario, and keep apprised of how well the student is performing the training task.

Report Generation enables the instructor to generate, via the system computer, a report of student or class performance, or the performance of students over several classes. The instructor can generate a report summarizing the results of statistical tests/measures of a student's performance in order to support feedback to the student. This information can assist the instructor in pinpointing areas of weakness in both the student and the training exercise.

Student Tutoring is a computer-based instruction capability that provides pre-programmed training exercises via the simulator's computer system. This capability allows the student to practice, usually in a self-paced fashion, pre-programmed training scenarios. Students can branch into remedial training for weak areas or delve deeper into areas of interest. This feature, therefore, provides an adaptive capability.

Training Exercise Selection and Training Exercise Creation/Modification (i.e., Training Exercise Control) is a capability that allows the instructor to perform one or more of the following: generate training exercises, select from a set of pre-programmed exercises, or modify existing training exercises. This provides the instructor with a great deal of flexibilty and control over the training environment. Also, it provides a technique for keeping training exercises updated and in line with changes in the actual system.

Student Sign-In is a capability that enables the student to identify himself/herself (usually for recordkeeping purposes) by entering a name or identification number into a file in the system's student monitoring software program. If it is intended that the simulator provide a means for recording, scoring, and/or storing student records for future use, then a sign-in capability is a necessary feature. This feature not only provides a means for establishing a unique repository for each student, but also provides a tracking function that allows the student to re-enter an instructional progression following a break or delay in the training sequence.

Malfunction Insertion/Selection is a necessary feature for MTSs which allows the instructor to create and/or select the malfunctions to be presented to the student during the training scenario. The instructor is able to insert pre-programmed malfunctions from a menu list, and often is able to create new malfunctions to meet new requirements.

Augmented Feedback is a training feature that provides the student with messages (i.e., knowledge of results) concerning the correctness of his/her input on a particular exercise. The message is usually presented via a video display screen. The comprehensiveness of the feedback can range from a buzzer indicating that an error has been committed, to a detailed explanation (text and graphics) of the error. A means for gradually reducing the feedback should be included for systematically reducing student dependency.

Next Activity Control enables the instructor to turn on or off the next activity pre-programmed for the student, or allows the instructor to select the next activity from a list of pre-programmed activities. As a result, the instructor can tailor a specific sequence of training scenario activities for a given student.

Cue Enhancement permits the highlighting (magnifying, intensifying) of stimuli or responses in order to draw attention to a particularly critical issue. On/off control of this feature should be available to the instructor.

System Parameter Control allows the instructor to set system parameter values prior to exercise commencement, or to input new system parameter values during a training exercise. Changes in parameters such as temperature, meter deflection, voltage, pressure, etc. can add to the challenge of a training scenario and let the instructor test the student's troubleshooting skills.

CONCLUSIONS

An analysis of survey data gathered on 51 maintenance training instructors revealed a "minimum" list of 13 critical functional capabilities that should be considered for implementation in maintenance training simulators. This list is by no means inclusive of all possible functional capabilities, but rather represents a common core of features (identified by knowledgable users) considered critical in supporting training effectiveness across most MTSs. The decision to add to this feature list for implementation of additional capabilities should be made on a case-by-case basis using information gathered during the front-end analysis phase of system procurement.

Diversification among MTS types will no doubt continue. Regardless of the multiple MTS configurations, however, certain critical functional capabilities should be designed into new systems. Furthermore, these functional capabilities should be implemented in a standard format across all MTS (to the extent possible) in order to reduce design costs, improve ILS through commonality, and promote transfer of user skills across training systems.

REFERENCES

Carroll, R.J., Evans, D.C., Sramek, J.B., Chiplock, L.W., Roth, J.T., & Dwyer, D.J. (in preparation). Maintenance Training Simulator Instructor Station and Student Station Design Specification and Guidelines. NAVTRASYSCEN 85-C-0017, Naval Training Systems Center, Orlando, FL.

Carroll, R.J., Thocher, L.I., Roth, J.T., & Massey, R.H. (1984). Maintenance Training Simulators: Their Use, Cost, and Perceived Effectiveness. AFHRL-TR-84-53, Air Force Human Resources Laboratory, Training Systems Division, Lowry Air Force Base, CO.

Hritz, R.J. & Purifoy, G.R. (1984). ISD-Based Maintenance Training Equipment Design Specification. AFHRL-TR-84-43, Air Force Human Resources Laboratory, Training Systems Division, Lowry Air Force Base, CO.

Nauta, F. (1985). Alleviating Fleet Maintenance Problems Through Maintenance Training and Aiding Research, NAVTRAEQUIPCEN MDA903-81-C-0166-1, Naval Training Equipment Center, Orlando, FL.

Naval Training Systems Center Instruction 4120.3D. (1984). Standardization of Components/Equipment Needed for Naval Training Devices. Naval Training Equipment Center, Orlando, FL.

ABOUT THE AUTHORS

Daniel J. Dwyer is a Research Psychologist in the Human Factors Division of the Naval Training Systems Center. He has over eight years experience in military training R&D. Specific work experience has been in the areas of human factors front-end analysis for maintenance training, development of design guidelines for maintenance simulators, evaluation of visual display parameters for training and aiding applications, computer-assisted instruction, and embedded training. Mr. Dwyer received his B.S. degree from SUNY Cortland in 1977 and his M.S. degree in Industrial/Organizational Psychology from the University of Central Florida in 1979. Currently, he is working on his Ph.D. in Industrial/Organizational Psychology at the University of South Florida.

Robert J. Carroll is a Principal Scientist with Applied Science Associates (ASA) in Butler, PA. He has over eight years experience in project management and human factors engineering, training systems analysis and design, and simulation. Mr. Carroll has successfully directed numerous projects both at ASA and while on the research faculty at North Carolina State University. Mr. Carroll holds a B.S. degree in Engineering Psychology from Tufts University and an M.S. degree in Psychology (Ergonomics) from North Carolina State University. Serving as a human factors consultant, Mr. Carroll has conducted human factors front-end analyses on a variety of systems ranging from the design of an innovative tape-to-braille display device to the Automatic Flight Control System of the CH-46 helicopter. Mr. Carroll has also served as a human factors consultant to Goodyear Aerospace Corporation and General Electric Corporation. He has developed detailed physical and functional requirements for the design of a training simulator for the CH-46 helicopter, and has performed research to standardize the design of maintenance training simulator instructor and student stations.

F-16 FLIGHT CONTROL SYSTEM TRAINING GAME

Lea T. Adams
Charles E. Thomas III
Honeywell Inc.
Systems and Research Center
3600 Technology Drive
Minneapolis, MN 55418

ABSTRACT

Research in diagnostics demonstrates that a critical difference between expert and non-expert technicians is experts have a good conceptual device model similar to the actual device structure while non-experts have inaccurate models generated from inferential misconceptions. Our goal was to bypass the novice-expert continuum by eliminating the novice's generation of misconceptions. Our approach was to develop a computerized adventure game whose underlying "world" was isomorphic to a specific device, (i.e., F-16 Flight Control Pitch Trim Subsystem [FCS]). Adventure game players develop maps or diagrams of adventure game environments. By taking advantage of this game strategy, novices can generate device structures by playing an adventure game with an environment isomorphic to the device. The statistical results of a pilot study showed that the adventure game training medium (1) facilitated learning of the structure, function and troubleshooting of the FCS, (2) decreased the probability of misconception generation, and (3) was a highly motivating learning environment.

INTRODUCTION

Overview of the Problem

Current maintenance trainers employ a procedural, step-by-step "cookbook" approach to system diagnostics. This approach focuses on teaching the use of technical orders (T.O.s). The device-specific structural knowledge and the troubleshooting strategies necessary for device maintenance are not explicitly taught or learned. As a consequence, when T.O.s fail, as they often do, technicians tend to resort to costly, inefficient, and ineffective "swaptronics" (i.e., blind removal and replacement of Line Replaceable Units).

A tour of training systems taught us some very important lessons. First, instructors want to teach their students three things. Instructors want their students to learn: (1) how to follow the T.O. procedures, (2) device-specific "theory" of operation, and (3) generalized troubleshooting strategies. Second, maintenance trainers were heavily criticized by instructors and students for an inability to teach anything but procedures. There is no "theory" of the device itself. And third, there is an expressed need for conceptual trainers that focus on the actual model of the device.

Research in the area of diagnostics demonstrates that a critical difference between expert and non-expert technicians is their detailed knowledge of the structure, connectivities, and functionality of specific devices [1]. That is, experts have formed a good working conceptual model of the device. Additionally, research suggests that an expert's mental model of a device or system is very close to the actual structure of that device or system. Non-experts, on the other hand, are continually forming inaccurate models based on misconceptions generated from inferential or incomplete data [2]. Trainers that focus on procedures alone encourage the generation of misconceptions because knowledge of the structure and function of the device can only be obtained through inferential processes. Memory retrieval research clearly shows that even when confronted with recognizably correct and disconfirming evidence, it is very difficult to disabuse a person of inaccuracies that they themselves have *generated* [3,4]. Misconceptions are thus very costly in time to remove.

Typically, training programs that address expert-novice differences have approached the problem by attempting to correct and expand a novice's mental model of the problem. We have developed an approach that allows technicians to learn the expert mental model of a device directly so that misconceptions are never formed. This training method sidesteps the typical novice to expert continuum by training directly to the expert's device model. This method involves the development of an adventure game environment for training.

Games as a Training Medium

Using games as a training medium is not a new idea. Many have exploited the concept of an educational "game" in order to generate and maintain interest in the to-be-learned topic [5,6]. In other words, games are considered great motivational tools for learning.

Research on the use of games as an instructional media indicates that a game must incorporate challenge, fantasy, and cognitive curiosity for it to be effective in an educational domain [7]. Research also suggests that a side effect of playing cognitive games is the development of cognitive (i.e., mental) models. This means that information presented within the realm of cognitive games is highly memorable and easily accessed because of the richness of the retrieval cues associated with the resulting mental model. An adventure game has all of these characteristics. It is an interactive game, usually involving fantasy, in which the player becomes personally involved through role-playing. The interactive aspect requires the player to take control and make decisions while his personal involvement increases the emotional investment in success. Some research is being conducted on the use of adventure games for training [8]. The focus of these adventure training games is their highly motivational and increased memorability qualities.

APPROACH

We believe the adventure game as a training medium has more qualities than simple motivation and enhanced memory for information presented during play. By utilizing every aspect of the adventure game genre, we have developed a powerful approach to the training of mental models of devices for maintenance technicians. First, because the underlying structure of an adventure game is a mythical "world," that the player travels through, that "world" can be constructed to be isomorphic to an actual device model. Second, strategies employed by adventure game players are analogous to very powerful psychological learning principles. For instance, adventure game players generally develop the strategy of drawing maps or diagrams or taking notes to aid in their orientation through the "world." This means that players are *generating* a representation of the world. Self-generation of a concept or model insures retention and access of that model. Recall that is exactly the reason why it is hard to disabuse a person of *self-generated* misconceptions. Also, the players are generating models in a representational form that is easily understood and salient to them. Not everyone can read, understand, or draw maps. Others cannot understand propositional information without an accompanying diagram. Because players generate the models as they think best, they are representing the information in the most comprehensible form for them. This also insures enhanced learning and promotes

reasoning about the "world." Third, adventure games generally require problem solving in order to progress through the environment. The puzzles and problems that need to be solved can be designed to utilize the general strategies required for eventual troubleshooting. Fourth, general characteristics of games, such as scoring, allow for immediate feedback regarding the correctness of actions. Therefore, instead of using an adventure game as a cognitive and motivational *medium* in which to present game-unrelated information, we turned the information to be learned into a complete adventure game. That is, it's the *game* students need to learn.

The technical approach was fairly straightforward. A subset of the F-16 Flight Control Pitch Trim Subsystem (FCS) was selected as the system to implement in this proof of concept development. Actually, different component types representative of pitch trim systems in general were selected in order to test this approach to training for various types of device components. Though the representation is not the verbatim F-16 FCS, the point we are trying to make is that whatever representation is put into this game is exactly the representation that people generate for themselves. Our system consisted of eight interconnected Line Replaceable Units (LRUs). Each LRU contained one to six subLRUs (functional components contained within a specific LRU). Individual signal pathways between LRUs and subLRUs were explicitly represented. This architecture was encoded on an IBM-PC in PASCAL and translated into a fictional world-like domain that was isomorphic to the actual system architecture (e.g., subLRUs became rooms and signal pathways became passageways).

Development

First, an editor that accepts as input a diagrammatic representation of a system was developed. This diagrammatic representation provides a coordinate system to the control program and also becomes the underlying "world" of the adventure game. Flexible data structures were developed that could represent the states of the FCS as well as the states of objects in the game itself. These databases allow the game procedures to become more generic.

In order to make gaming analogous to the system's functions, the experimenters had to first determine how general pitch trim subsystems worked. During this phase, we realized that the pitch trim subsystem was best presented in terms of large functional pieces; each piece had a logical motivation for its existence. That is, there was a "backbone" structure that reflected the system's most basic function. There are then subsystems that are "added" to this backbone to insure this basic function under certain circumstances, such as during take-off or when the autopilot is engaged. There is research evidence to support the notion that presenting device information in this fashion is beneficial because it allows a trainee to understand and chunk information in an efficient manner. Therefore, we broke the system into its backbone structure and three subsystems. To date, only the backbone structure and the first subsystem are implemented. A diagram of the backbone structure can be seen in Figure 1. Figure 2 is the backbone structure and the first subsystem. After players progress through the backbone structure, the world expands (e.g., previously locked doors can now be opened) to include the next logical subsystem.

The result was a way of training the conceptual representation of a device without revamping existing procedural trainers. We developed an adventure training game of a Flight Control Pitch Trim System that runs complete on an IBM-PC. It can supplement current procedural trainers and can be used by trainees as on-site, off-site, or off-duty training.

EVALUATION

The system was evaluated quantitatively. Six subjects were run in a pilot study. Five of these subjects were completely unfamiliar with Flight Control Systems. Four of the subjects had never played computer adventure games before. Two of them had never had any experience with computers.

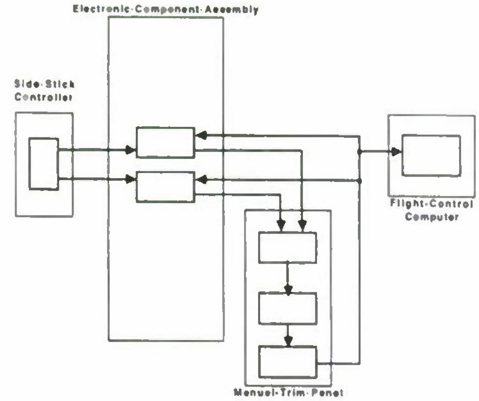


Figure 1. Structural Backbone of the F-16 Pitch Trim Subsystem

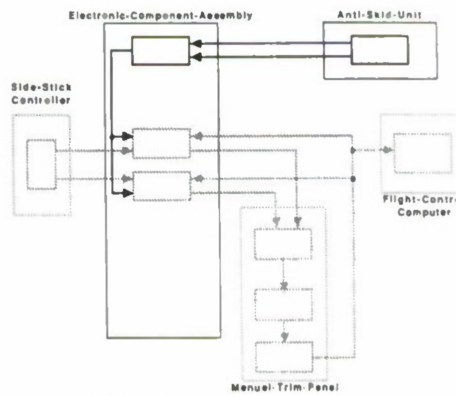


Figure 2. On- Ground Pitch Centering Subsystem

The procedure was as follows. First a pretest to measure prior knowledge of the F-16 FCS was administered. (This pretest consisted of 14 questions, half of which dealt with structure and connectivity and half of which dealt with inferencing and reasoning about the system.) This test was then split in half into two versions, Version A and Version B, each with seven questions. These were used as post-tests and counterbalanced for order. The subject then played the F-16 adventure game. The amount of playing time varied from one hour to one hour and 25 minutes. All subjects had to play until they moved through the five critical subLRUs of the backbone structure and play at least one hour to insure experiential equivalence. Subjects were encouraged to take notes or "something" to keep track of their progress. Upon completion of the game, subjects' maps or notes were taken away and a post-test was given (Version A or Version B). The maps or notes were then returned to the subjects and a second post-test was given (the version not given for the first post-test).

We analyzed accuracy of pre and post-test performance and also performed a descriptive evaluation of the subject-generated representations. The mean percent correct on the pretest for the six subjects was 28.57. The mean scores for the two post-tests, without the use of notes and with the use of notes, were 45.24 and 54.76, respectively. *T*-tests were performed between pretest and post-test 1 performance and pretest and post-test 2 performance. Neither *t*-test was statistically significant, though the latter was approaching significance, $t(5) = 148, p > 0.05$ and $t(5) = 1.69, p > 0.05$, respectively. Upon examination of the data, it was noticed that wherever Version A of the post-test was administered (post-test 1 or post-test 2), subjects' performance was greater than on Version B. This suggested that the two versions of the

post-test were not of equivalent difficulty. Therefore, data from the two post-tests were collapsed and a t-test between the pretest and the post-test was performed. The results of this test showed significant improvement from pretest to post-test, $t(5) = 2.32$, $p < 0.05$.

Subjects' representations were also examined for errors or generated misconceptions. No errors were found. Subjects expressed great interest in the game and most voiced their disappointment when instructed to cease play.

CONCLUSIONS

The pilot study, even though there was a small number of subjects, revealed some interesting results. First, scores improved significantly from pretest to post-test, suggesting significant learning occurred after only one hour of play. Additionally, subjects were able to answer questions regarding troubleshooting the system, even though they were only directly trained to the structure. That is, subjects were accurately reasoning and inferencing about the system. Second, the representations generated by the subjects were accurate, though they differed in form from one to another. This suggests a decreased probability of misconception generation common to current procedural trainers. And last, all subjects protested when asked to stop playing the game. It appears that the adventure game medium is a highly motivating and stimulating learning environment.

We are attempting to obtain permission to field test this approach with actual flight line technicians in order to incorporate their feedback into the future development of this training technique. Also, this approach is currently being evaluated for possible patent by Honeywell's Systems and Research Center.

REFERENCES

1. Gittomer, D. H. (1984). *A cognitive analysis of a complex troubleshooting task.* Unpublished doctoral dissertation, University of Pittsburgh.

2. Bransford, J. D., Vye, N. J., Adams, L. T., and Perfetto, G. A. (in press). *Learning skills and the acquisition of knowledge.* In R. Glaser & A. Lesgold (Eds.), The Handbook of Psychology and Education. Hillsdale, NJ: Erlbaum.

3. Adams, L. T. (1986). *Facilitating spontaneous access in problem solving.* Unpublished doctoral dissertation, Vanderbilt University.

4. Perfetto, G. A., Bransford, J. D., and Franks, J. J. (1983). Constraints on access in a problem solving context. *Memory & Cognition,* 11 (1), 24-31.

5. Dodge, B. J. (April 1980). *Learner interest and instructional design: A conceptual model.* Paper presented at the Annual Convention of the Association for Educational Communications and Technology, Denver, CO.

6. Dunn, J. J. (1984). *Gaming approaches in Educational Software: An analysis of their use and effectiveness.* Available from ERIC Document Reproduction Service, Arlington, VA.

7. Malone, T. W. (1980). *What makes things fun to learn? A study of intrinsically motivating computer games.* (Report CIOS-7 [SSL-80-11]) Palo Alto, CA: Xerox PARC.

8. Williams, F. D. (October 1984). *Educational Adventure Games.* Paper presented at the Annual Conference of the North American Simulation and Gaming Association, Iowa City, IA.

ABOUT THE AUTHORS

Dr. Adams and Mr. Thomas, Senior Research Scientists at Honeywell's Systems and Research Center, have collaborated on several innovative approaches to training in military applications.

Dr. Adams has a Ph.D. in Experimental Cognitive Psychology and has conducted research in the areas of schema and script theory, novice-expert differences, knowledge organization and structure, and the relationships of knowledge structures to acquisition and retrieval of information. She has authored and coauthored several major publications in these areas. Her dissertation addressed spontaneous access of relevant information during problem solving.

Mr. Thomas holds an M.B.A. and an M.A. in Educational Psychology. He has been involved in a wide variety of Honeywell research and development projects involving AI JPA technology, instructional analysis, and human factors. Mr. Thomas has coauthored papers on intelligent training systems, training, knowledge-based job performance aids, and human factors in space.

APPLICATION OF EXPERT SYSTEM TECHNOLOGY TO AID CONTROLLER/ROLE PLAYERS IN A HIGH REALISM TRAINING ENVIRONMENT

By Gary J. Pasewark, Staff Engineer
The Singer Company
Link Simulation Systems Division
11800 Tech Road
Silver Spring, Maryland 20904
(301) 622-8740

ABSTRACT

Many current command and control training devices use a role player concept. In this concept the target students interact with the device through personnel who play the role of superior, adjacent, and subordinate groups. The role players receive information from the training device and communicate it to the student staff as they would in real life. The credibility of the information flow to the student staff is as dependent on the role players as it is on the fidelity of the device. Problems arise from excessive role player workload, role player gamesmanship and the use of personnel with minimal training as role players. These problems increase as the complexity of the training requirements increase.

The solution to these problems is to provide the controller/role players with aids to ease their workload and allow them to concentrate more fully on responsibilities that their played roles require. One such aid is the application of expert systems technology.

INTRODUCTION

This paper reports on the status of an R&D program at the Link Simulation Systems Division of The Singer Company which is aimed at providing controller/role players with aids to ease their workload through the application of artificial intelligence expert systems technology. The expert system approach will automate low level decisions and allow the role player to monitor, override when appropriate , and maintain a high level of realism in the complex environments of command and control training.

ROLE PLAYER CONCEPT

The role player concept is used in many current training systems in the Army, Navy, and Marine Corps. This concept separates the target students from the training device. Figure 1 shows an example of this concept. In the arena

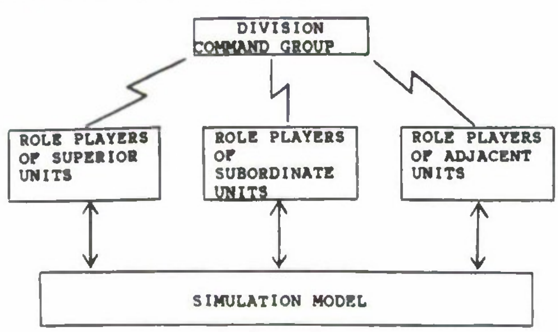


FIGURE 1. THE ROLE PLAYER CONCEPT.

of command and control, the personnel targeted for training don't fight the battle directly, they control the battle. Through the use of communications and situation analysis, command and control personnel must direct subordinate personnel to fight the actual battle, while coordinating with adjacent and superior units. The targeted students use the equipment that would be available in a real situation. All groups which interact with the student staff are role played by personnel who receive the information from the actual simulation device.

Controller/Role Players carry out the following tasks:

- Act as the primary interface between the student staff and the simulation
- Play role of assigned personnel
- Communicate and coordinate with student staff
- Communicate and coordinate with other role played staffs
- Monitor battle situation relative to the assigned role
- Make control decisions over subordinate units
- Issue orders to subordinate units thru the system interactors

As an example, lets look at the role players required for the Army Training Battle Simulation System (ARTBASS). This device, designed and developed by Link, is designed to exercise a Battalion command group (battalion commander and his staff). The role players represent the subordinate, adjacent, and superior personnel that interact with the Battalion staff. These are as follows.

- Maneuver Company Commanders (2 to 4 present)
- Fire Direction (Artillery Support)

- Air Liaison (Air Support)
- Brigade Staff (S1, S2, S3, S4)
- Engineering Support
- Maintenance Support
- Administration and Logistics Support

The role players interact with the simulation to issue orders and receive feedback of events and results from action. They then communicate events to the student staff using the standing operating procedures (SOP) of the unit. The role players for the maneuver company are normally the actual company commanders from the battalion that is being exercised.

The stressful environment created for both the command group and maneuver company role players adds to the realism of the training for the command group and provides secondary training to the maneuver company commander (company commanders are "lone ranger" commanders who have no staff and are actually involved in fighting the battle). The role player concept works well in this application with a high fidelity model providing platoon information to the role player.

Problems may arise in the use of the role player concept for any of several reasons.

- The role player assignment may be too broad.
- A given simulation may require the role player to make too many manual decisions for subordinate units.
- The role player may not be prepared for the assigned roles.
- The role player may exercise excessive gamesmanship (the desire for the role player and staff to "look good" at the expense of training fidelity or realism).
- The diversity of role player's abilities may lead to variation of training effectiveness.

The first three items above lead to excessive role player workload. The problems of excessive workload becomes more evident in higher echelon devices where role players represent functions which would be carried out by an entire staff rather than by one individual. Two methods have been used to alleviate the role player workload. First, reduce the complexity and fidelity of the simulation by using aggregate modeling approaches. Second increase the number of role players to spread the workload. Both of these have negative effects on the device. The first reduces the realism of the information flow and can create a negative training situation. The second approach causes the cost of use of the device to go up drastically.

The best approach to alleviate excessive role player workload has two parts. First, the man/machine interface must be easy to understand and must provide all the information that the role player needs and in a format that lends itself to the tasks of role playing. Second, the simulation device must provide realistic responces for the independent actions of simulation assets under the control of the role player. When these capabilities are added to command and control training devices at echelons above Battalion, the result will be increased realism, reduced overwork of role players and reduced cost.

The concepts of expert system technology are well suited to the tasks of automating low level decision for simulated assets.

APPLICATION CONCEPT FOR EXPERT SYSTEM TECHNOLOGY

Many areas of application of expert system type control exist in an Army command and control environment. Most support control and low level tactical decisions can be reduced to heuristic rules (expert rules of thumb).

The basic concept of a role player aid using expert system technology is shown in Figure 2.

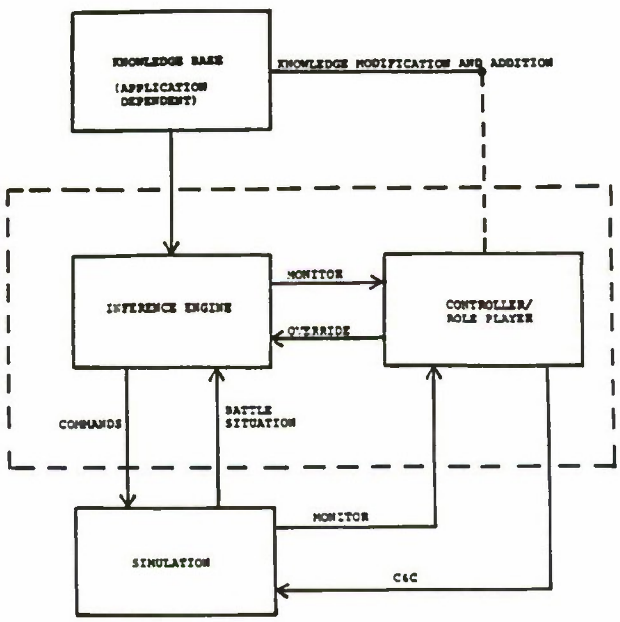


Figure 2 Expert System Control

The simulated unit's tactics and standing operating procedures (situational response) are coded as heuristic rules in the knowledge base. The inference engine compares battle situation evidences with the knowledge base to trigger commands to the unit. In this way, the expert system frees the role player to carry out tasks of communication and high level decision while the low level actions and response are carried out automatically. The system also allows the role player to monitor the action taken by the expert system and override them as he deems appropriate.

This application of expert systems technology has several requirements which are not met with commercially available expert systems shells. Some of these special requirements are as follows:

- Multiple assets under simultaneous control (multiple friendly units under control).
- Interaction and coordination between controlled assets (certain tactical responses require several units to coordinate actions toward a particular results).
- Representation of heuristic rules and procedural control.
- Evidence arrays (such as "range to detected target", for each detected target).
- Real time processing (within the real time context of the host simulation).

After studying commerically available expert system shells it was determined that each of these special requirements goes beyond their capabilities. With the clear need for instructor/role player decision aids in mind we decided to create our own expert system shell customized to these special requirements. Working with the CRT Corporation as consultants, Singer initiated the development of a new expert system shell. Our design goal was to create a fast expert system shell that would provide the capabilities needed for our applications. With this in mind we decided to avoid the features that slow down many existing shells. Several features were dictated by our special requirements:

- A rule base that is fully compiled before run time, (no lexical analysis at run time)
- Implement expert system shell in a fast military specifications approved language (Fortran)
- No dynamic space allocation or garbage collection
- Operation pruning of if clauses
- Forward chaining inference mechanism
- Allow evidence arrays
- Support multiple assets under simultaneous control
- Knowledge base divided into multiple rule lists grouped by function. Use hierarchy of rule lists to limit checking of unnecessary rules.

The tactical rule language developed was an attempt to provide a method for tactical domain experts to represent rules and limited procedural control. The following features were designed into the tactical rule base language:

- An integrated, set based representation
- Distinctive syntactic structure for
  - Evidence declaration
  - Derived evidence construction
  - Assertion construction
  - Rule definition
  - Operator and action definition
- Intrinsic textual elaboration as part of the syntax
- Simple if/then relationship between antecedent conditions and consequence
- Indentation to illustrate control structures
- Evidence arrays and loop structure
- Increased readability of rules by allowing two name for clauses (i.e., target-in-range and target-not-in-range)
- A syntax that supports complex logical relationships, fuzzy logic, arithmetic computations, etc.
- Explicit use of parentheses around operators and parameters in conditions to eliminate the problems of procedure and associativity.

EXPERT SYSTEM PROTOTYPE

The actual prototype application of the expert system technology to aid role players required a decision as to what task to choose. Current research carried out by Link on Division level command and staff training indicates that many role play areas would benefit from expert system aids. Our experience with the Army training community has indicated mistrust of systems making automatic tactical decisions for simulated combat units. Because of this we decided to concentrate our efforts on the many support areas required by a Division in combat.

Other current research being carried out at Link centered on adding models to support Division level training. One model being tested was for Intelligence and Electronic Warfare (IEW). This model simulates the activities of ground based electronic warfare (EW) assets. This was a good candidate for the application of a role player aid because of the speed of response necessary to react to enemy radio use. A role player will have difficulty monitoring and controlling electronic warfare assets without aids to help carry out low level decision and activities covered under the units standing operating procedures (SOP). The tactics rule base controlling the electronic warfare assets is a collection of heuristic rules representing the standing operating procedures (SOP) and operator judgements and procedures. The standing operating procedures

contain electronic target priorities for listening, attacking (with artillery or air), or jamming. Figure 3 shows the decision process in EW operations. Electronic warfare operations are carried out by the subordinate units that act to intercept enemy messages, identify enemy nets in use, and locate the position of the enemy. Most situations that arise are covered

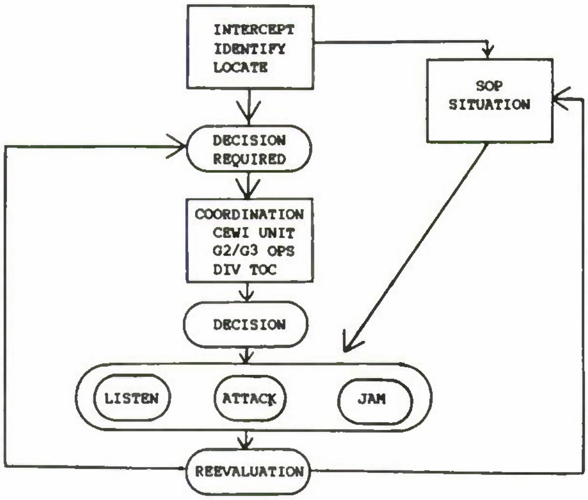


FIGURE 3. DECISION PROCESS IN EW OPERATION

by the units standing operating procedures (SOP). Those situations that are not in the SOP require decisions by the division staff (the primary students being exercised). The center route in Figure 3 shows this situation. The expert system would automate the actions of the subordinate units as they intercept, identify, locate, and apply the SOP to the situation. The controller/role player would act to monitor the subordinate units and report to and coordinate with the division staff being exercised. The expert system would automate the top two blocks of figure 3, that is, the rules and procedures necessary to intercept, identify, and locate an ememy radio net and the SOP situation analysis necessary to decide what to do once it is found.

For the expert system to operate, the system extracts evidence data from the host Division simulation. The evidence data represents the EW assets situation and the battle situation which would normally be observed by a role player. The evidence is then used to test the tactics rule base defining the normal response necessary to control the EW asset.

The actions commanded by the expert system are then passed back to the host division simulation similar to an order issued by a role player.

This application of expert systems technology has many development advantages over many other real time uses of expert systems. Since the expert system does not totally replace the role player, but is designed as a aid, the rule base does not need to cover every situation. A role player is required to monitor, make high level decisions, and communicate to the target staff. The system does alleviate excessive workload and reduce the total number of role players required.

By using the expert system to aid role players in their work the tactics rule bases can be built up in complexity over time while the system is being used to help the role player produce more realistic information flow.

SUMMARY

High realism training devices that place a strain on instructors or controller/role players can benefit from the addition of expert system aids. By placing emphasis on the solution of a problem and ignoring application purity, expert system technology can be used to meet the requirements of a rigorous real time training environment.

The application of expert system technology can provide the training community with several advantages.

- Decreased cost by reducing controller/role player requirements
- Decreased variability of training by providing a minimum response level for the control of simulated units
- Increased realism of information flow provided to the student staff by freeing the controller/role players to concentrate on communication and high level decision making
- Reduced controller/role player gamesmanship by automating low level decision making

REFERENCES

- Kaplin, I.T. "Evaluating the Human Role in MSG Systems" Fifth Annual Course on Modeling, Simulation and Gaming of Warfare, Georgia Institute of Technology 1984.

ABOUT THE AUTHOR

Mr GARY PASEWARK is a Staff Engineer with Link Simulation Systems Division of the Singer Company. He holds a BS degree in Physics and Mathematics from the University of Maryland and has over ten years experience in real time simulation for training. Since joining Link in 1976, he has been involved in training device developement for ASW training, Naval tactical command and control training for submarine and surface vessels, and Army command and control training. He is currently principle investigator of Link's Independent Research and Developement project entitled "Combat Command Training/Expert Systems for Joint Military Operations".

KNOWLEDGE-BASED SIMULATION: AN APPROACH TO INTELLIGENT OPPONENT MODELING FOR TRAINING TACTICAL DECISIONMAKING

Azad M. Madni, Ph.D
Perceptronics, Inc.
21111 Erwin Street
Woodland Hills, CA 91367-3713

Robert Ahlers, Ph.D
Human Factors Division, Code 711
Naval Training Systems Center
Orlando, FL 32813-7100

Yee-yeen Chu, Ph.D
Perceptronics, Inc.
21111 Erwin Street
Woodland Hills, CA 91367

ABSTRACT

Modern weapon systems have greatly expanded the range of options that can be exercised by trained tactical decisionmakers. However, tactical training environments today are unable to create the different opponent behaviors necessary to challenge the decisionmaking skills of tactical commanders. Since the use of human opponents is clearly not cost-effective, this training requirement falls under the purview of computer-based simulation. This paper presents a knowledge-based simulation approach for tactical adversary modeling along with an interactive user interface that allows non-programmers to modify simulation models on-line. A laboratory application that addresses a set of training objectives appropriate for surface warfare officer training is also included. The suggested approach is directly applicable to meeting current training simulation requirements generated by the surface Navy. Both the simulation approach and the software implementation are upward-compatible with the modeling of coordinated adversaries and supporting team members.

INTRODUCTION

The ability of modern weapon systems to engage multiple, sophisticated targets is providing a strong impetus for incorporating similar capabilities into related training environments. However, incorporating the different opponent behaviors for challenging modern tactical decisionmakers requires additional sophistication in today's tactical training environments. The use of human controllers continues to be cost-ineffective. Currently, instructors and device operators cannot efficiently represent realistic behavior of multiple targets. This is due to many factors, including conflicting training responsibilities, time overloading and varied familiarity with enemy tactics for the broad range of platforms generally required for training. Consequently, this requirement is a good candidate to be addressed by software models. However, there are several drawbacks to current software implementations of tactical threats/targets. These are outlined in Table 1.

TABLE 1
CURRENT IMPLEMENTATION DRAWBACKS

- Current software implementations of automated targets
  - do not represent decisionmaking behaviors of enemy tacticians (no planning functions; only reaction to situation)
  - do not take into account training objectives of trainee
  - involve lengthy software update cycle

Recent developments in Artificial Intelligence (AI) and cognitive modeling make it possible to develop knowledge-based techniques for modeling and controlling the opponent's behavior. This paper describes such an approach. It presents an intelligent opponent model that responds to trainee's actions in accord with training goals and inferred trainee deficiencies. Specifically, the opponent model consists of both domain-dependent knowledge and general problem-solving heuristics, thus allowing it to initiate and control tactics, maneuvers, and subsystem operations.

KNOWLEDGE-BASED SIMULATION OF TACTICAL ADVERSARIES

Enemy platforms that exhibit complex, purposeful behaviors are necessary to challenge tactical decisionmakers. However, the behavior of automated targets, has, heretofore, been driven primarily by analytical models or by a table of pre-enumerated behavior options, resulting in opponent behaviors that are either purely time and trajectory based, or capable of reacting to the prevailing tactical situations only with little provision for proactive tactical training.

Knowledge-based expert systems technology has opened up a new dimension in tactical simulation in general, and tactical adversary simulation in particular. Specifically, expert systems technology can be harnessed to model and emulate the behavior of an enemy commander who is in tactical control of an adversary platform. In other words, knowledge-based expert systems technology can be exploited to simulate an "embedded enemy tactician." The capabilities inherent in the use of knowledge-based expert simulation of adversary behaviors are summarized in Table 2.

An analysis of the tactical decision making task reveals the following important features:

- While tactical engagements do have generic phases, tactical behaviors tend to be opportunistic, subject to satisfying tactical doctrine and rules of engagement.
- Incoming information goes through a process of fusion and aggregation prior to producing useful intelligence.

- Tactical time horizons produce high time-stress and mental burden.

TABLE 2
ADVANTAGES OF KNOWLEDGE-BASED ADVERSARY SIMULATION

- Controllable computer-based opponent behavior
  - reasonable tactics and doctrines
  - rules of engagement
- Inspectable behaviors
  - decisionmaking processes
  - rationale/assumptions behind decisions
- Provision for realtime software update and knowledge base maintenance
  - opponent knowledge base
  - friendly characteristics knowledge base
  - scenario knowledge base
  - terrain, weather, geopolitical database
- Usable by computer-naive tactical domain experts and instructors
  - graphical (iconic) interfaces with graphical editors
  - "what if and why" facilities

The tactical environment is generally characterized by time-varying data and event-driven tactical strategies and decisions. The tactical decisionmaking task can be simply characterized as a real-time problem that requires continuous reasoning in the face of time-varying, uncertain data, changing constraints and evolving objectives. To this end, the one desirable approach to implementing controllable intelligent opponents is the use of a modeling framework that represents the task knowledge in a modular, hierarchical, conceptually transparent manner.

A particular class of expert systems architectures that is well-suited for modeling human performance in tactical decision making tasks is the "blackboard model" [9, 10]. The blackboard model is both a problem-solving metaphor and a particular rule-based modeling framework appropriate for problems requiring continuous "real-time" reasoning under poorly defined conditions. Based on the metaphor of "experts" sitting around a blackboard, the blackboard model coordinates the activities of a number of different knowledge sources (i.e., "experts" or "specialists") by providing a global data base (i.e., blackboard) among them. The knowledge sources (KSs) cooperate with each other via the blackboard. The partial solutions that are produced by the knowledge sources to the problem under consideration are posted on the blackboard.

The blackboard itself is divided into a number of abstraction levels, each containing hypotheses for partial solutions which are linked to other levels by logical relationships. A monitor process controls access to these hypotheses and inspects any changes to notify "interested" knowledge sources.

In the blackboard model, each KS is independent of the other KSs, i.e., no KS knows which or how many other KSs exist. In general, a KS monitors a specific level of the blackboard for those changes or conditions that are relevant to its area of expertise. When these changes or conditions occur, the KS requests a processing turn by placing an event-related item on the agenda. This agenda is a list of possible processing events from which the scheduler chooses the one most likely to lead the farthest in solving the problem. The decisionmaking process of the scheduler is controlled by an invariant set of rules about problem solving and a dynamic goal structure. As the solution progresses, the goal structure adapts to focus attention in a data-directed fashion. The chosen processing event is passed back to its KS for execution

The blackboard idea has been extended to hierarchical models in which different concepts are assigned to different blackboards and knowledge sources mediate information among the blackboards in the form of expectations, supports, refutations, etc. Nii [14] has presented a summary of blackboard models and applications.

NAVAL SURFACE WARFARE APPLICATION

The knowledge-based simulation approach has been used to develop an intelligent adversary simulation within the context of Naval surface warfare. The simulation scenario is staged against the backdrop of ongoing limited hostilities between a Blue and a Red Naval vessel. It commences with the Tactical Action Officer onboard a Blue Cruiser maintaining watch. Intelligence assets have located the Red fleet in the vicinity of the Blue cruiser. The specific assignment for the Blue Cruiser is a Red Destroyer that has been designated a "prime target."

The TAO onboard the Blue Cruiser has been given specific guidance and is assumed to be aware of the local rules of engagement (ROEs). Table 3 provides illustrative ROEs for this application.

TABLE 3
EXAMPLE OF LOCAL RULES OF ENGAGEMENT

- Attack to disable/destroy any prime target that commits a hostile act
- Hostile acts include enemy behaviors such as:
  - Fire Control System emissions associated with surface-to-surface missile (SSM) for a period greater than or equal to 1 minute
- Attack prime targets only if there is a high certainty that prime targets have:
  - Identified Blue capabilities and
  - Identified Blue location and
  - Have engaged in concerted attack
- Concerted attack implies that more than 2 SSMs have been launched against Blue platform

The intelligent opponent simulation is constructed against the foregoing scenario backdrop. The simulation employs the blackboard model-based problem-solving software architecture presented earlier. The specific instantiation of this architecture is shown in Figure 1.

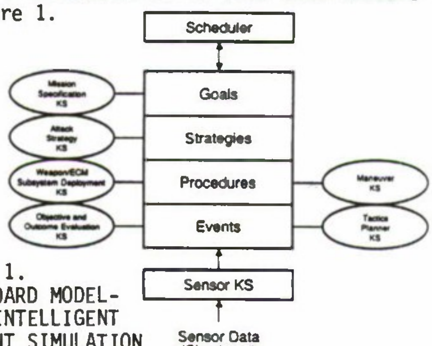


FIGURE 1.
BLACKBOARD MODEL-BASED INTELLIGENT OPPONENT SIMULATION ARCHITECTURE

As seen in Figure 1, several different KSs are involved in representing the knowledge of the tactical opponent. Based on the discussions with Navy Tactical Officers, these KSs are partitioned into tactically meaningful classes that roughly correspond to the generic categories of situation assessment, objective formulation, tactics planning and tactics execution. Some examples of the tactical rules associated with the KS are given in Table 4. Note that the rules consist of a set of conditions and actions or procedures. Details of the software architecture are described in Chu and Shane ([5]).

TABLE 4
EXAMPLES OF TACTICAL RULES

IF	THE BLACKBOARD CONTAINS A NODE: TRY CHANGING SPEED, and THE OBJECTIVE IS: TO NEGATE A POSSIBLE ATTACK
THEN	TAKE PRECAUTION BY MOVING FASTER.
IF	THE OBJECTIVE IS: TO NEGATE POSSIBLE ATTACK and IT HAS BEEN FOCUSED ON FOR AT LEAST 4 CYCLES
THEN	POST INFERENCE NODES ON THE BLACKBOARD TO i) TRY CHANGING HEADING and ii) TRY LAUNCHING A CHAFF
IF	THE BLACKBOARD CONTAINS A NODE: TRY LAUNCHING A CHAFF, and THE FOCUSED EVENT IS: TO NEGATE A POSSIBLE ATTACK
THEN	if RED SHIP CAN TURN BEFORE A MISSILE COULD ARRIVE then MAKE AN ACTION: LAUNCHA CHAFF
IF	THE FOCUSED ACTION IS : LAUNCH A CHAFF
THEN	COMPUTE THE BEST HEADING FOR RED TO TURN, CALCULATE THE CHANGE IN HEADING TO BEST HEADING (HEADING-CHANGE) if HEADING-CHANGE = 0 then TAKE ACTION TO LAUNCH CHAFF, if HEADING-CHANGE ≥ 20 then ADJUST RED'S HEADING TO BEST HEADING (with high confidence) and LAUNCH CHAFF if HEADING-CHANGE ≤ 20 then ADJUST RED'S HEADING TO BEST HEADING (with little confidence) and LAUNCH CHAFF
IF	A BLUE'S MCS IS ACTIVATED
THEN	POST INFERENCE NODES ON THE BLACKBOARD i) OBJECTIVE IS: TO NEGATE A POSSIBLE ATTACK and ii) TRY CHANGING RED'S SPEED

INTELLIGENT OPPONENT BEHAVIOR MODIFICATION/UPDATE

For training tactical decisionmaking, it is necessary that the student be exposed to a host of opponent behaviors. To this end, the training system contains interface facilities for the instructor, or other subject matter expert, to modify the adversary's tactical rule base in support of required training objectives.

The tactical behavior of the platforms used in training simulations cannot be "coded and forgotten." Not only does our knowledge about opposition tactics change, but the tactics themselves also change. In addition, the emphasis that is placed on specific aspects of the tactics also changes. This results in a requirement to create tactical targets that are not necessarily "tactically accurate" but which are designed to fulfill or complement specific training objectives. To this end, an instructor is provided with interface facilities for straightforward creation or modification of tactical targets.

To provide proper interface facilities, we chose to use the direct, graphical manipulation capabilities to allow an instructor to specify variations of data and procedures that govern an adversary's behavior. Figure 2 shows a sample screen of the knowledge base browsing mode that supports this process. The opponent's tactical knowledge at the highest level is represented using Modified Petri Nets (MPNs), ([11]; [12]) based upon the Petri nets formalism ([15]; [1]). MPNs are used to model opponent task execution at those levels of abstraction where there is a high

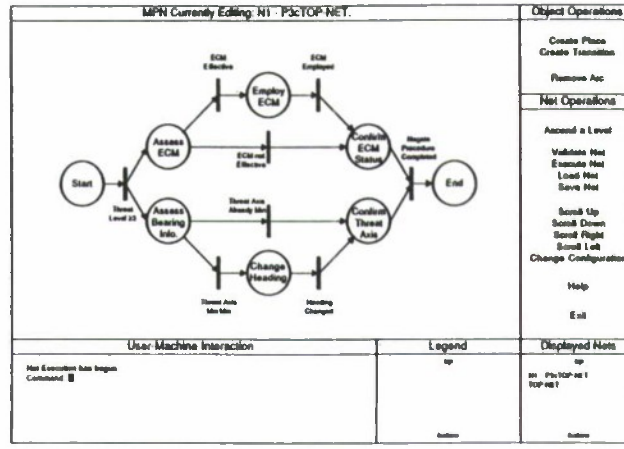


FIGURE 2.
GRAPHICAL INTERFACE FOR VIEWING/EDITING OPPONENT'S "NEGATE ATTACK PROCEDURE"

degree of concurrent task-related activities and tactical decisions ([20]; [11]). An MPN is graphically represented by two types of nodes: places (circles) and transitions (vertical bars). The nodes are connected by directed arcs (arrows). The places, transitions and directed arcs represent the static properties of the network. The dynamic property is represented by a token (dot) that resides in a place. When a token resides in a place, the activity associated with the place is said to be "ongoing." When the event associated with the output transition occurs (i.e., the transition "fires"), the ongoing activity ceases and the token propagates across the transition to the output places (i.e., activities). Thus, the control knowledge associated with a specific intelligent adversary procedure can be parsimoniously represented with the MPNs framework.

The declarative knowledge in the opponent simulation is represented using frames ([13]). Specifically, the approach employs the Frame Representation Language ([17]) which features multiple inheritance paths between frames. A frame is associated with each place and transition in the network. Procedural knowledge is embedded within specific frame slots using production rules. The separation of control knowledge from the declarative and procedural knowledge provides an executable opponent simulation that is straightforward to generate and easy to inspect and modify. This multi-formalism approach to representing opponent simulation knowledge makes for a parsimonious, executable tactical expert model. The graphics interface uses multiple windows and a mouse pointing device to allow the instructor to browse a knowledge base. Instructors can also view the execution of specific tactics during real-time opponent decisionmaking in much the same way a student using GUIDON-WATCH ([16]) can view the reasoning process during diagnostic problem-solving.

The rule-based representation of KSs provides a convenient basis for constructing an interface for modifying tactical rules. This interface provides the necessary flexibility for the instructor to modify the expertise and rules used by the simulated tactical adversary. In other words, this capability is constructed as a knowledge acquisition interface that allows the expert instructor to specify variations of rules and procedures that govern an adversary's behavior. These modifications result in alternative adversary performance models. Use or avoidance of particular tactics or systems, and the

creation of particular tactical situations to which a trainee must respond, are all accomplished via the knowledge base interface facilities.

Current interface facilities allow the user to browse through and selectively review portions of an adversary's tactical rule sets including rules of engagement (ROE) and rules for situation assessment and planning.

Current Capabilities of the Intelligent Adversary Simulation

The current capabilities of the intelligent opponent simulation are summarized in Table 5. The trainee has total control of the Blue platform via the trainee station. Specifically, the trainee can select tactics, specify maneuvers and deploy weapons or sensory assets. The trainee maintains situation awareness by monitoring the geopolitical situation, the various subsystems, and the message window, or alternatively by querying the appropriate knowledge bases via the commands menu.

TABLE 5
CURRENT CAPABILITIES OF
INTELLIGENT ADVERSARY SIMULATION

- Trainee can control Blue ship
  - Tactics
  - Maneuver
  - Subsystems
- Trainee monitors
  - Geopolitical situation
  - Subsystems
  - Commands
  - Message window
- Computer opponent can control Red ship
  - Tactics
  - Maneuvers
  - Subsystems, including missiles, sensors and countermeasures
- Effects on World Model of
  - Red maneuver, tactical moves and deployment of assets
  - Trainee's decisions and "non-decisions"

The automated opponent has total control of the Red ship's tactics, maneuvers and the various subsystems including missiles, sensors and countermeasures. The manner in which the opponent controls these assets can be changed interactively by the instructor in the knowledge base browsing/editing mode.

All actions taken by the trainee and the intelligent computer opponent update the world model. For example, Red maneuvers, tactical moves and deployment of assets all update the world view. Similarly, the trainee's decisions and "non-decisions" in terms of use of friendly assets, and the execution of specific tactics also update the world model.

The prototype intelligent adversary was done on the Symbolics 3670 using the Heuristic Control System, a rule-based derivative of AGE [2] system developed at stanford University. A simulated environment of one-on-one engagement was also developed to codify and test expert rules for tactical planning. The combined system [5] has demonstrated that (1) the rule-based representation of tactical knowledge provides a convenient basis for constructing the interface facilities for modifying the tactical rules of the computer adversary, and (2) the blackboard model-based architecture provides an effective framework for representing tactical decisionmaking behavior. In simulation exercises, the modification of adversary parameters can be based on a library of instructional outlines designed to focus attention of the trainee on the appropriate skills prior to playing the modified scenarios. In this setting, the trainee is able to exercise and tune specific target skills individually and in relevant combinations. It appears that such a training, authoring and delivery environment is not only motivating to the user but can also be expected to greatly shorten the software update cycle involved in supporting specific types of tactical training exercises.

SUMMARY

Future Research Directions

Future research will involve additional work in regards to three major themes: the graphical interface that the computer-naive instructor was to specify/modify tactical targets, the simulation of multiple coordinated adversaries, and the simulation of supporting teams.

Intelligent Opponent Simulation Language. The case has been made above for having an interactive interface for specifying/modifying the characteristics and behavior of tactical opponents. The extensive use of graphics and abstract symbology at varying levels of abstraction is expected to make the interface more natural to use, and the targets even easier to access and modify. The graphical interface provides the necessary flexibility for the instructor/experimenter to modify the expertise and rules used by the simulated tactical adversary.

Multiple Coordinated Adversaries. The work described above for single targets will be extended to allow the simulation of opposing forces consisting of multiple platforms. The modeling of such composite forces is extremely complex since the behavior of each platform must be specified in terms of its relationships to all other platforms within the force. The complexity of each individual model therefore is also a function of the constraints that result from the need to maintain coordination with other platforms. To achieve the goals of field implementation and online instructor modification of tactical targets, requires a method for reducing this complexity. Work in AI has addressed the distributed decisionmaking problem [3; 6; 7; 8; 18; 19] and will be applied to the implementation of a generic mechanism for dealing with the multiple constraint environment of coordinated tactical adversaries.

Simulation of Supporting Teams. The products from the above two efforts (i.e., the graphical interface for tactical targets and multiple coordinated adversaries) will be applied to the simulation of supporting teams. For example, in submarine combat systems team trainers, the activities associated with the sonar, radio rooms, and helm control are now performed by training device operators. A development effort will be undertaken to simulate these teams' activities in order to attempt to reduce the manning requirements for the trainers associated with the current and next generation fire control systems. The approach will again be to provide a high-level graphical interface for implementing the actual models. The goal will be to provide not only the simulation models but also the tools to allow the models to be updated in the field.

Conclusions

Current implementations of automated targets are deficient because they do not represent the decision making behaviors of enemy tacticians. Specifically, automated targets react to the prevailing tactical situation only; they do not engage in higher level tactical planning functions. The approach taken in this paper addresses this deficiency by modeling the decision making behavior of the opposing tactical commander. In addition, an interface is described which has been designed to facilitate the maintenance of the opponent's tactical rule bases. Plans are underway to extend this overall approach to the modeling of coordinated tactical adversaries and supporting teams.

REFERENCES

1. Agerwala, T. Putting Petri Nets to Work. Computer, pp. 85-94, December 1978.

2. Aiello, A., Bock, C., Nii, H.P., and White, W.C. AGE Reference Manual, Technical Manual. Computer Science Department, Stanford University, CA, October 1985.

3. Cammarata, S., McArthur, D., and Steeb, R. Strategies of cooperation in distributed problem-solving. In Proceedings of the Eighth International Joint Conference on Artificial Intelligence, pages 767-770, The International Joint Conferences on Artificial Intelligence, Karlsruhe, West Germany, August 1983.

4. Chu, Y. and Madni, A.M. Network-Based Human Behavior Modeling for Adaptive Training. Proceedings of 1985 International Conference on Systems, Man, and Cybernetics, Tucson, Arizona, November 1985.

5. Chu, Y. and Shane, D. Intelligent Opponent Simulation for Tactical Decision Making. Final Report, PFTR-1120-11/86, Perceptronics, Inc., Woodland Hills, CA, December 1986.

6. Corkill, D.D. An Organization Approach to Planning in Distributed Problem Solving Systems. Technical Report TR 80-13, Department of Computer and Information Science, University of Massachusetts, Amherst, Massachusetts, May 1980.

7. Dalkey, N.C. Group Decision Making. Technical Report UCLA-ENG-7749, School of Applied Science, University of California, Los Angeles, July 1977.

8. Davis, R. A model for planning in a multi-agent environment: steps toward principles for teamwork. Working paper 217, Massachusetts Institute of Technology AI Laboratory, October 1981.

9. Erman, L.D., Hayes-Roth, F., Lesser, V.R., and Reddy, R.R. The Hearsay-II Speech Understanding System: Integrating Knowledge to Resolve Uncertainty. ACM Computing Survey, 12: 213-253, 1980.

10. Hayes-Roth, B. The Blackboard Architecture: A General Framework for Problem Solving? Technical Report HPP-83-30, Heuristic Programming Project, Computer Science Department, Stanford University, May 1983.

11. Madni, A.M. and Chu, Y. Network-Model-Guided Expert Knowledge Elicitation, Proceedings of the 1985 IEEE International Conference on Systems, Man, and Cybernetics, Tucson, Arizona, November 1985.

12. Madni, A.M., Chu, Y., Purcell, D. and Brenner, M. Design for Maintainability with Modified Petri Nets (MPNs): Shipboard Propulsion System Application. Perceptronics Report PFTR-1125-84-11, prepared for Office of Naval Research, November 1984.

13. Minsky, M. "A Framework for Representing Knowledge," in the Psychology of Computer Vision, P.H. Winston (ed.), McGraw-Hill Book Company, New York, 1975.

14. Nii, P.H. Blackboard Systems: The Blackboard Model of Problem-Solving and the Evolution of Blackboard Architectures (Part One), The AI Magazine, Summer 1986.

15. Peterson, J.L. Petri Net Theory and the Modeling of Systems, Inglewood Cliff: NJ: Prentice Hall, 1980.

16. Richer, M.H. and Clancey, W.J. GUIDON-Watch: A Graphics Interface for Viewing a Knowledge Base, Report No. STAN-CS-85-1068, Department of Computer Science, Stanford University, August 1985.

17. Roberts, R.B. and Goldstein, I.P. The FRL Primer, Report AIM-408, Artificial Intelligence Laboratory, MIT, Cambridge, MA, July 1977.

18. Rosenschein, J.S. and Genesereth, M.R. Communication and Cooperation. Technical Report HPP-4-5, Heuristic Programming Project, Computer Science Department, Stanford University, October 1984.

19. Smith, R.G. and Davis, R. Frameworks for cooperation in distributed problem solving. IEEE Transactions on Systems, Man, and Cybernetics, SMC-11(1):61-70, January 1981.

20. Zisman, M.D. Use of Production Systems for Modeling Asynchronous Concurrent Processes in D.A. Waterman and F. Hayes-Roth (eds.), Pattern-Directed Inference Systems, Academic Press, New York, NY, 1985, pp. 53-68.

ABOUT THE AUTHORS

Dr. Azad Madni is a Senior Vice-President and Chief Scientist of Perceptronics. He received his Ph.D. from the University of California, Los Angeles in Engineering and Computer Science, specializing in Human-Computer Systems. His current research interests are in intelligent tutoring systems, cockpit automation, adaptive decision aiding, and intelligent man-machine interfaces. He is a member of IEEE, AAAI, AIAA and ADPA.

Dr. Robert Ahlers is a Research Psychologist with the Human Factors Division of the Naval Training Systems Center. He received his Ph.D. from North Carolina State in Human Factors. His recent work has addressed the use of artificial intelligence techniques to provide performance diagnosis and tactical adversary capabilities.

Dr. Yee-yeen Chu heads the Knowledge Systems Group at Perceptronics. He received his Ph.D. in Mechanical Engineering with emphasis in human-computer interaction from the University of Illinois. His research interests are in intelligent tutoring and aiding systems, robotics, and intelligent interfaces. He is a member of IEEE, AAAI and ACM.

THE TRAINING OF EXPERTS FOR HIGH-TECH WORK ENVIRONMENTS

Sherrie P. Gott, PhD
Robert Pokorny, PhD
Air Force Human Resources Laboratory
Manpower & Personnel Division
Brooks AFB, Texas 78235-5601

ABSTRACT

When a training program fails to markedly influence the development of high-tech complex skills (such as electronic troubleshooting), the failure can generally be traced to two sources. First, failure occurs when training is not based on clear and explicit models of the desired expertise. For problem solving expertise, specifications of the expert's internal strategic processes for handling complex problems and the particular forms of knowledge and skill that support the strategies are especially critical. Secondly, failure occurs because the training of complex mental skills often fails to consider the conditions that are needed for the development of cognitive expertise, though similar conditions for the development of advanced physical skills are well known. They include extensive, constructive practice sessions where "the game is played" (i.e., authentic problems are solved) under realistic conditions. For such practice to be constructive, the trainee needs commentary and guidance from a coach who, among other things, can model the desired (problem solving) performance and carefully sequence problems according to the trainee's progress, while at the same time providing external support in the form of problem solving hints and instructional information. This set of conditions requires the learner to adopt an active role in skill development and situates learning and extended practice in the context of real world problems. This instructional approach is in contrast to traditional, more passive skill training where the instruction amounts to telling students <u>about</u> a domain such as electronics rather than providing learning experiences for <u>doing</u> electronic problem solving.

A large research and development program is underway in the Air Force to train technicians for complex work environments in a manner that seeks to avoid these pitfalls. The Air Force Basic Job Skills (BJS) Research Program is examining the performance of technical experts in dozens of occupations to establish models of expertise as targets for training. Advances in knowledge engineering procedures such as those used in developing expert systems are being applied to specify in great detail the technical expert's strategies and supporting skill and knowledge bases. Of particular interest are dimensions of expert performance that cut across Air Force jobs and can thus be characterized as <u>basic</u> to expertise in complex work environments. In some sense these common dimensions can be viewed as modern day basic skills or the skills needed for a technologically advanced world. In addition, applications of artificial intelligence to instruction in the form of intelligent tutoring systems are being utilized to create the desired conditions for active, problem-oriented learning. In this paper, work done with over 15 experts in four related electronic and computer maintenance jobs will be highlighted to illustrate the "engineering" of expert knowledge. Also, a successful training study conducted with apprentice electronic technicians will be reported. In this study, the standard obstacles in complex skill training were satisfactorily overcome.

KNOWLEDGE ENGINEERING FOR INSTRUCTIONAL APPLICATIONS

Intelligent tutoring systems (ITS) offer the kinds of learning conditions that are thought to be important in the development of complex cognitive skills, i.e., guided practice in realistic problem solving. They require at least three types of knowledge bases as their infrastructure. First, there is the knowledge that constitutes expertise in the domain being taught. This is the expert model, which represents the goal state for trainees or the set of ideal performances the instruction should produce. Secondly, dynamic information about what a trainee knows and doesn't know is necessary to model student performance during learning. Since student performance data is needed so that instructional decisions can be made, some investigators have suggested that this knowledge base is best conceived as a layer of the third ITS information structure, the curriculum. Curriculum knowledge of course provides the subject matter content and instructional treatments that are intended to move students toward the goal of expertise as represented by the expert model. Accordingly, information about a student's performance and understanding may best be expressed in terms of his/her status with respect to curriculum goals and subgoals, e.g., proficiency in schematic tracing. A knowledge engineering methodology designed to generate these elements for the training of complex technical problem solving (e.g., electronic troubleshooting) has been developed as part of the Air Force Basic Job Skills Research Program. The methodology and illustrative results are described below.

<u>Cognitive Task Analysis Methodology</u>

The approach to knowledge engineering in the BJS effort involves real-time problem solving, multiple stages and types of knowledge

engineering inquiry, and a number of formats for knowledge representation, some of which have been adapted from knowledge engineering work in medical diagnosis.

In the first stage of the process, hands-on technical experts in a particular AF specialty generate a set of authentic problem scenarios that are representative of all types of problems, i.e., faults encountered on their equipment systems. In the second stage, pairs of experts pose the scenarios to each other so that their work performance can be realistically sampled. (The expert who poses the problem knows the location of the fault; the expert who attempts to solve the problem does not.) During the solution process, the researcher probes the expert solver to establish the series of actions s/he executes in solving the problem. Reasons for the actions, interpretations of outcomes resulting from the actions, and block diagram-like sketches of the equipment affected by the actions and outcomes are recorded as well. This part of the knowledge engineering process corresponds to the generation of sequences of mental events called PARI structures (Precursor [to Action]-Action-Result-Interpretation). Comparable frameworks have been used in the engineering of medical diagnostic knowledge. (1) An example of PARI data is shown in Table 1 for a single action node.

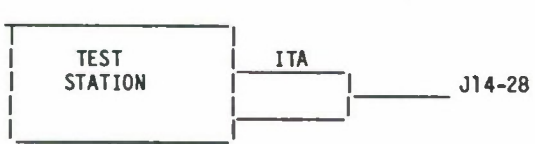
TABLE 1: PARI DATA

Precursor: Want to see if the stimulus signal is good up to test package cable.

Action: Measure signal at J14-28 with multimeter

Result: 28 volts

Interpretation: This is expected reading; this tells me that the stimulus is getting from the test station through the cable, so that part of the stimulus path is good



In the third stage, a series of rehashes occurs during which the researcher probes the expert for various kinds of information to elaborate and extend the PARI data, including the following: alternative results that could be expected as outcomes for a given action and interpretations of such alternative results; alternative actions to satisfy the same goal (as stated in the precursors); alternative precursor-action pairs that would be reasonable to pursue; reasons to support the selection and sequence of goals (precursors); reasons to support the expert's preferred actions; and finally, the specific knowledge and skills required to carry out each PARI sequence. Examples are shown in Table 2.

TABLE 2: Rehashes of PARI Data

(1) The technician is asked what other result(s) would be expected as an outcome to each action, and what that result would reveal.

Alternative Result 1: 0 volts

Alternative Interpretation 1: the problem is upstream from this measurement point; since the output is 0, the problem could likely be in a connection, or in the stimulus generator.

Alternative Result 2: 18 volts

Alternative Interpretation 2: the problem is upstream from the measurement point; since the output is low rather than 0, the problem is more likely to be in some component rather than in a connection.

(2) The technician is asked to generate alternative actions to satisfy the same goal (as stated in the precursor):

Precursor: Verify stimulus signal is good up to test package cable.

Alternative Action: Measure stimulus signal output from test station entering test package cable and then swap cable.

(3) The technician is asked to generate alternative precursor (goal) - action pairs that would be reasonable to pursue, e.g.,

Alternative Precursor: Want to see if measurement signal path is good.

Alternative Action: Insert a signal from a known good generator to the beginning of the measurement path.

In the final stage of the process, the focus of analysis shifts from PARI sequences for a single problem to the full complement of troubleshooting problems for a given job. The goals are to consider all instances of actions, goals (precursors), system diagrams, and supporting reasons and then classify redundant and related instances into appropriate categories. Tables 3 and 4 illustrate this process for actions and precursors. Once a skill category, such as "measurement taking," emerges, it becomes the basis for the final representation of a troubleshooting knowledge component, namely, a skill definition. The collection of skill definitions for a job domain serves the more detailed function of describing exactly how procedures are executed for purposes of instruction and assessment.

Making procedural skills clear enough for teaching and testing purposes requires an analysis of the subcomponents of the skill and the conditions under which the skill is activated. For example, measurement taking involves knowing the signal's expected value and type, selecting and operating the measurement device, and reading and interpreting the measured property. These subcomponents are apparent in

Table 1 PARI data. Procedural subcomponents and conditions provide a framework for generating progressively harder instances of the skill. For example, easy to hard instances of conditions calling for the selection of a measurement device would be elicited from the expert and then utilized as curriculum knowledge. The predetermined sequence would provide input to instructional planning.

Similarly, the three knowledge bases required for intelligent tutoring systems are provided: expert problem solving performance data represents the domain expertise; the problem set plus elaborations and skill definitions constitute the instructional content; and problem solving performance data for less-than-expert technicians both informs student modeling and highlights expert-novice differences in ways that suggest instructional tactics.

TABLE 3: Grouping/Classification of Action Instances (Procedures/Operations Component)

Action Instance	Procedural Category
-check pins on test package -check fault indicator light	• visual inspections
-swap Threat Simulator A5 card (probable cause of failure) -swap card N4A1 with like N4A3 card	• swapping
-run diagnostics on high frequency measurement card and coax switch -run diagnostics with bit dump	• computer control/software interpretation
-test for good signal out of TG4 with oscilloscope -ohm check between J110 and J4 with digital multimeter (DMM) -put N4A1 on extender and test for -18VDC with DMM	• measurement taking

TABLE 4: Grouping/Classification of Precursor Instances (Strategic Knowledge Component)

Precursor Instance	Goal Structure Category
- want to verify 5V power supply fail not a fluke - want to verify failed diagnostic not a fluke	• verify fail
- need more information on drawer serviceability - need more information on resources used in failed test - want to trace stimulus input to get complete routing	• expand information on probable cause of failure and its inputs/outputs
- want to test most likely suspect on stimulus path - want to check other cards in signal flow - want to check for good input signal at N4A1 (probable cause of failure) - want to check wiring between source of signal (N3A16 card) and N4A1 (probable cause of fail)	• test input/output suspects to probable cause of failure

Fine-grained problem solving information such as that described above is ultimately aggregated, summarized, and abstracted across problems and across experts' (and novices') solutions to provide a coherent statement about technical performance for a given job. Such characterizations constitute the aforementioned explicit models of expertise that determine the effectiveness of complex skills training.

Knowledge Engineering Results

Approximately 15 experts and 200 less-skilled technicians in four related AF electronics specialties have participated in knowledge engineering studies as described above as part of the Basic Job Skills Research Program. On the basis of these studies, a meaningful superstructure for organizing troubleshooting

performance data has been developed: It consists of three major components: (1) system knowledge or the equipment device models used in problem solving (e.g., system knowledge regarding stimulus or measurement functionalities); (2) troubleshooting procedures or operations performed on the system; and (3) strategic knowledge, which includes (a) strategic decision factors that involve fault probabilities and efficiency estimates and (b) a top-level plan or strategy component that is responsible for component orchestration in task execution. The orchestration occurs as the Strategy component which sits on top of the Procedures and System Knowledge components deploys pieces of knowledge and procedural subroutines as needed and as driven by the decision factors (Figure 1).

System Knowledge. In this cognitive skills architecture (Figure 1), system knowledge provides the dominant organizing principle. For example, the System Knowledge component provides the foundation to which the companion Procedures component is attached. According to this view, a measurement or swapping operation is attached to a device model representation, since the purpose of the operation is viewed as adjusting the present model of the device with new knowledge of faulty components. Similarly, an information gathering procedure such as software interpretation is directed toward elaborating the available device model with instantiations and details relevant to the particular problem. System knowledge also feeds the strategic decision factors that underlie the strategy component, since these factors involve system fault probabilities and efficiency estimates associated with operations on the system. Finally, system knowledge influences the goal structure of the general strategy component in the sense that certain areas of the equipment are targeted before others (again due to fault probabilities and efficiency considerations).

Procedures/Operations. During problem solving, expert electronics technicians adjust their model of system operation by performing two major classes of actions. The first class involves troubleshooting operations performed on the system, and the second consists of information gathering procedures that use external sources of system information. Action statements in PARI sequences provide the raw data source for this component.

Troubleshooting operations such as running a test or making a measurement typically refine the expert's belief about the location of the fault. In effect, the operations mark some portions of the equipment system as more suspect than

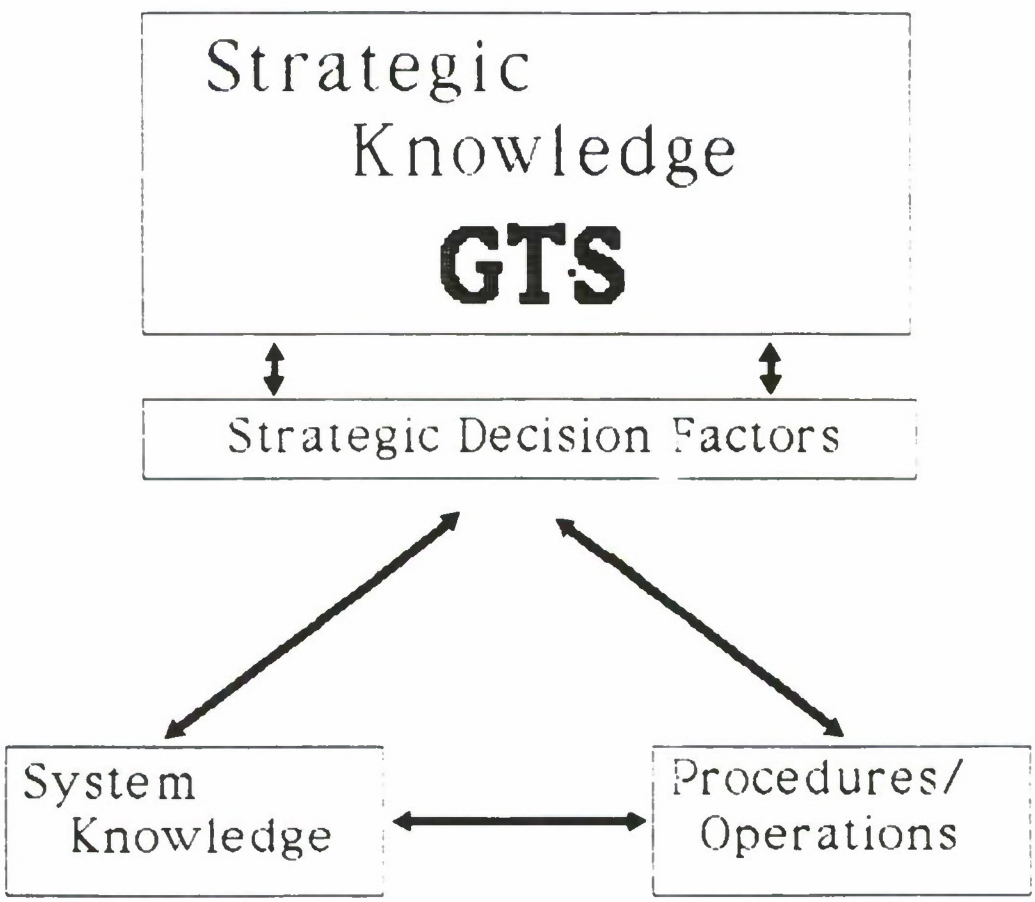


FIGURE 1: Cognitive Skills Architecture

others. The expert can then focus attention on the suspect components and elaborate a localized device model by either remembering more precisely how the suspect sections work or by learning about its operation by consulting external information sources. These sources include technical data or documentation sources such as schematic, wiring, and block diagrams; computer software; and the actual physical equipment itself.

Strategic Knowledge. Finally, there is the strategic component of the architecture shown in Figure 1, with its underlying strategic decision factors. When knowledge engineering is conducted for developing an intelligent tutoring system, it is particularly important to make explicit the factors that experts consider in deciding (a) how to sequence their fault isolation goals and (b) which troubleshooting operation or procedure to use in pursuing a specific subgoal. Data from the BJS project plus related research in troubleshooting suggest that these factors are based on three fundamental principles--probability, cost, and benefit.

Probability refers to the likelihood that a certain system component is defective. One kind of probability factor is the base rate of failure for a component. One section of the equipment may be more suspect than other equipment sections simply because one component generally breaks more often than the others. A second probability factor involves the association between system components and particular symptoms. For example, a zero volts fail makes connections more suspect than if a low voltage reading had been obtained. The latter would have implicated devices rather than connections.

Cost decision factors, or the obstacles in performing troubleshooting operations, can be represented as follows:

- Time: the longer an operation takes, the less preferred it is by the expert.

- Danger: the more dangerous an operation is (either to an operator or to the equipment), the less preferred it is.

- Dollars: the more expensive an operation is, the less preferred; e.g., repairing a component is favored over replacing it.

- Mental energy: the more mentally demanding an operation is, the less preferred.

- Physical energy: the more physically demanding, the less preferred.

Benefit decision factors primarily involve the quantity and quality of the information gain. Tests that generate more information about the signal path, for example, are favored over less informative tests. Experts prefer measurement over swapping for this reason, among others. Tests that are more reliable are favored as well, and so swapping may be preferred over a diagnostic self test having known unreliability.

To summarize, a primary source of failure in complex skills training programs, namely, deficient models of the targeted expertise, has been attacked analytically in the Air Force BJS research effort. A methodology that blends techniques from knowledge engineering and cognitive task analysis has made explicit the unobservable mental processes of expert troubleshooting and produced specific expert models to use as targets of instruction. Results have shown that the prototypical troubleshooting performances of AF technical experts in four related high-tech specialties are a function of three major classes of mental events - strategic knowledge, system understanding, and procedural skills interact in elaborate ways as the complex decision making involved in fault isolation unfolds. These characterizations of expertise are in turn treated as the expert models for intelligent tutoring systems which provide interactive learning environments capable of overcoming the other major obstacle to complex skills training. The curriculum and student knowledge bases required by intelligent tutoring systems can also emerge from knowledge engineering output of this type. A training study involving instruction which was developed in this way is described next.

A SUCCESSFUL TRAINING STUDY

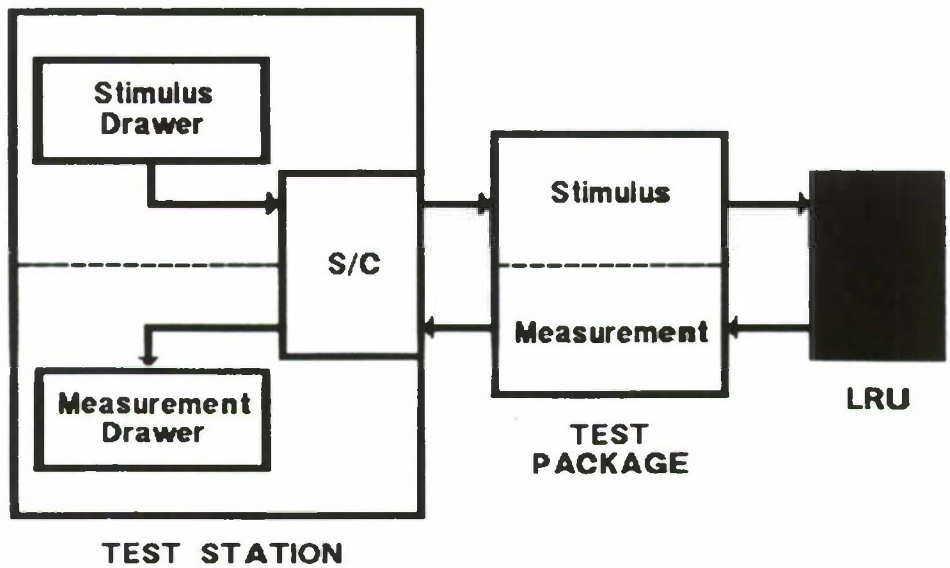
As a precursor to a BJS intelligent tutoring system to teach troubleshooting, a training study involving a human tutor (versus a computer coach) was conducted. The domain was F15 integrated avionics maintenance at the intermediate level of repair (automatic test equipment). The three ITS knowledge bases described above were established via cognitive task analysis and utilized in the instruction.

System (device model) knowledge, troubleshooting procedures/operations, and strategic knowledge were engineered for a group of expert, novice, and intermediate level avionics technicians on a constrained set of fault isolation problems. The expert-like skills targeted for enhancement were particular instantiations of the Figure 1 cognitive skills architecture. The system knowledge of interest was an abstracted characterization of the avionics system signal path, plus several layers of elaborated system knowledge. As shown in Figure 2, the path reveals the stimulus and measurement functionalities of the equipment system. The signal originates in the stimulus drawer of an avionics test station, travels through the station's switching drawer (S/C) which performs signal switching functions, and through an interface test package to an aircraft line replaceable unit (LRU) which is being tested for a malfunction. It returns through the interfact package to a measurement source in the test station. The procedures of interest were three methods for investigating the equipment that ranged from rudimentary to advanced:

(1) swapping equipment components
(2) using self-diagnostics to test system integrity
(3) measuring device and circuit functionality

Increasingly complex system and strategic knowledge are associated with increasingly sophisticated methods. For example, the swapping model draws upon superficial system knowledge where suspect components need only be known in the nominal sense, whereas detailed and functional device models are needed to support the measurement model. The self-diagnostics model requires knowledge of the relative reliabilities and information gains associated with the available self-diagnostic tests, whereas

FIGURE 2: **Avionics Equipment Configuration (Signal Path)**



reasoning about the costs and benefits associated with swapping procedures tends to be relatively simple. In addition to detailed system knowledge, the measurement model also requires complex decision making regarding where and how to take measurements as well as supporting skills to identify test points (schematic tracing) and to interpret measurement outcomes accurately.

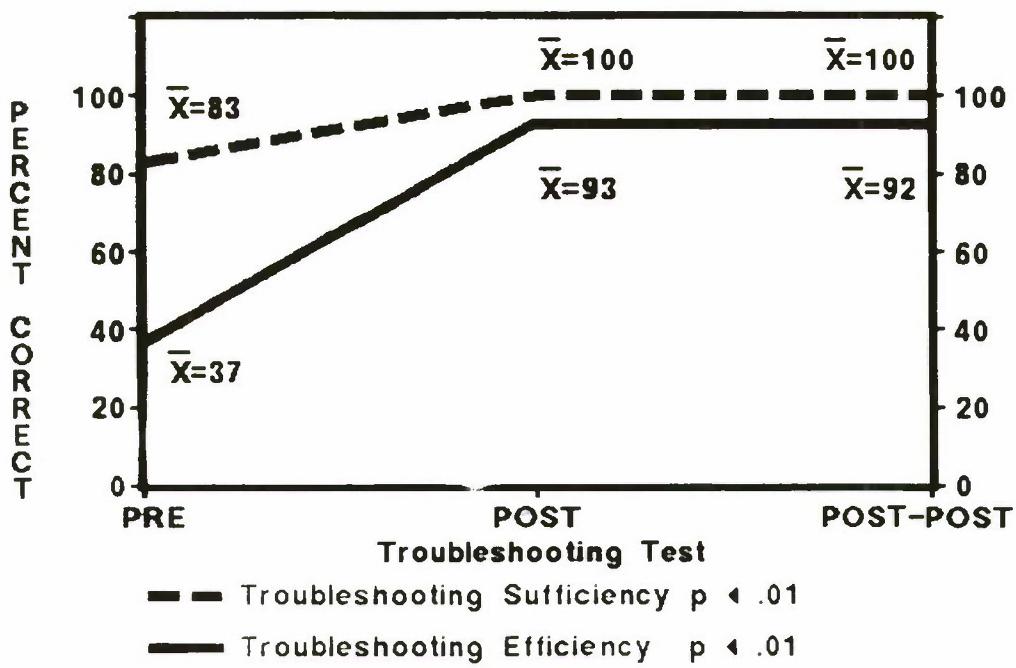
The treatment in the training study involved the posing of authentic troubleshooting problems similar to those generated in a BJS knowledge engineering study as described above. During three to five hours of individual instruction over a period of three days, seven technicians were tutored. They were presented a troubleshooting scenario and then probed regarding what they would do to isolate the fault (Actions), why they would take the particular action (Precursors), and what the outcome (Result) of the action meant to them (Interpretation). In effect, technicians were instructed to generate PARI records (see Table 1) including the associated device model sketches. The human tutor gave feedback to their stated Precursors, Actions, and Interpretations in the form of hints intended to move them toward more expert-like performances.

To evaluate their learning, they were given both an end-of-training problem-based test as well as a delayed posttest after the weekend. The tests were authentic troubleshooting scenarios belonging to the same class and difficulty of problems on which they had been tutored. Their progress was scored both in terms of the sufficiency of their operations--that is, whether they sufficiently investigated all suspect pieces of the equipment--and the efficiency of their moves--that is, whether they efficiently conserved time and equipment resources.

Results showed statistically significant improvements in both areas, with particularly dramatic gains in efficiency. Mean scores are plotted in Figure 3. The group's Sufficiency in examining all suspect parts of the equipment improved from a pretest mean value of 84 (range = 60 to 95) to a posttest mean of 100. The delayed posttest mean was also 100, indicating the improvement was retained over the weekend. The group's Efficiency in fault isolation improved over twofold. The mean pretest value was 37 (range = 24 to 52); the initial posttest mean was 92 (range = 81 to 100); and the delayed posttest mean was 93 (range = 81 to 97).

Pedagogically, this human tutor training study was based on the same instructional input and principles that underpin the avionics intelligent tutoring system. Expert models of performance provided the ideals used as instructional goals. Less-than-expert performance data provided the basis for a curriculum progression. MOre specifically, students were tutored along a progression of fault isolation methods that ranged from swapping to diagnostic testing to measuring. The progression was based on execution difficulty as demonstrated by the range of technicians studied in the task analysis phase of the work. Finally, the instruction took place in an active, problem-oriented learning environment designed to foster comples skill acquisition. Technicians were afforded extensive proctice in fault isolation; they were required to articulate and focus on their reasons and their interpretations of various troubleshooting moves; they were aided by a human tutor who principally through Socratic dialogue, challenged them to reflect on what they did in terms of expert standards of throughness and efficiency. The technicians later attributed the gains they made to the opportunities they had to practice fault isolation procedures intensively and solve problems independently. They reported that recording and reflecting on their actions and reasons was helpful and that they profitted from the hints and consistent feedback.

FIGURE 3: **BJS Proof of Concept Training Study**



This successful study is viewed as empirical support for the effectiveness of skill acquisition treatments that are driven by explicit expert models and characterized by learning conditions vital to the development of cognitive expertise. A more substantial proof of concept will occur during late 1987 when a BJS avionics intelligent tutoring system providing 50 hours of instruction will be operationally tested in three AF workcenters [2,3].

ABOUT THE AUTHOR

Dr Gott is a senior research scientist and Project Director of the Air Force Basic Job Skills Research Program at the Air Force Human Resources Laboratory. She holds an MA and PhD in Educational Psychology and has ten years experience in psychological research and development. Her current research interests are in the investigation of complex problem solving skills in advanced technical domains. More specifically, her work since joining AFHRL in 1979 has centered around the application of new analytic and instructional approaches coming out of basic research in the cognitive sciences (including artificial intelligence research) to the skill assessment and training problems facing the Air Force.

Dr Pokorny is a research scientist on the AF Basic Job Skills Program. He holds a PhD in Experimental/Cognitive Psychology and is currently interested in knowledge engineering techniques for use in developing complex skills tutoring systems.

AUTOMATED FLIGHT TEST DATA CORRELATOR FOR A HELICOPTER FLIGHT TRAINING SIMULATOR

Susan C. Garing and Michael L. Brychcy
The Singer Company — Link Flight Simulation Division
Binghamton, New York

ABSTRACT

This paper discusses an accurate, semi-automated method for increasing the performance fidelity of a helicopter flight training system's aerodynamic model. The method employs an automated correlation algorithm known as AUTOCOR for systematic adjustments of the quasi-static mathematical model using fundamental aerodynamic model parameters. The AUTOCOR algorithm is divided into two phases. The first concerns calculation of the incremental forces and moments necessary to modify the vehicle's static trim attitudes and pilot control positions to match those of the actual helicopter throughout its entire flight maneuver envelope. The second phase centers on optimal incorporation of these forces, moments, and other empirical adjustments into the simulator's model data tables by judicious use of numerical techniques. The AUTOCOR algorithm provides satisfactory results even with an incomplete flight test data set.

INTRODUCTION

There is no question of the importance of aerodynamic fidelity for the successful simulation of a full-mission helicopter. Correct aircraft flying qualities, both static and dynamic, are a fundamental training systems building block. The level of acceptable fidelity for military helicopter simulators has, in the past, been judged largely by pilot evaluation and tailoring. However, pilot tailoring of simulator math models is imprecise by its nature and thus a potentially time-intensive and unbounded task.

Recognizing this, increased emphasis on matching extensive aircraft static and dynamic flight test data (FTD) shows a potential for eliminating subjective tailoring for Army helicopters, much as many commercial aircraft simulators are developed today.

The current development of the Singer-Link Black Hawk full-mission simulator represents a program which reflects the continuing trend towards FTD correlation of the simulator. This simulation program has a requirement for matching an extensive FTD set (greater than 2,000 points), while still including pilot evaluation in regions where flight test data is unavailable. As shown in Table 1, this data covers the flight envelope of airspeed from -45 to +160 knots, density altitude from sea level to 15,000 feet, and gross weight from 15,100 to 22,000 lbs (external stores support system configuration).

FLIGHT TEST	ALTITUDE RANGE (ft)			SPEED RANGE (knots)		
STATIC LONGITUDINAL STABILITY	2,000	TO	10,000	88	TO	152
STATIC LATERAL STABILITY	2,000		10,000	55		124
CONTROLLABILITY - ALL AXES	2,400		6,800	45		140
MANEUVERING STABILITY	3,500		11,000	100		132
DYNAMIC STABILITY - ALL AXES	2,000		7,300	0		100
LEVEL FLIGHT PERFORMANCE	2,600		14,000	40		160
CONTROL TRIMS	400		10,000	-45		155
				±45 KNOTS SIDE FLIGHT		
HOVER PERFORMANCE (IGE AND OGE)	2,100		12,000			
CLIMB PERFORMANCE	0		15,000	0		83
AUTOROTATION ROD VS AIRSPEED	5,000		6,000	40		130
ROD VS MAIN ROTOR RPM	5,000		6,000	40		130

IGE = IN GROUND EFFECT
OGE = OUT OF GROUND EFFECT

TABLE 1 BLACK HAWK FLIGHT TEST DATA (FTD)

The AUTOCOR computer program was developed in response to these expanded requirements. This two-phase interactive program assists in identifying aerodynamic model deficiencies and in making the necessary analytic and empirical corrections to the model.

The fundamental principles of this technique are documented, with examples, by Hazen[1]. In brief, Hazen iterated upon incremental net vehicle forces and moments until the desired static aircraft attitude and control positions were obtained. These adjustments were then manually incorporated into the aircraft aerodynamic model as a smooth function of appropriate variables. For example, one might match desired fuselage pitch attitude by introducing an empirical pitching moment about the center of gravity as a function of airspeed. These forces and moments are illustrated in Figure 1. The AUTOCOR program expands this general approach by adding flexibility to sources of the necessary force and moment increments, and mechanization to the process of modifying the model data that characterizes these needed increments.

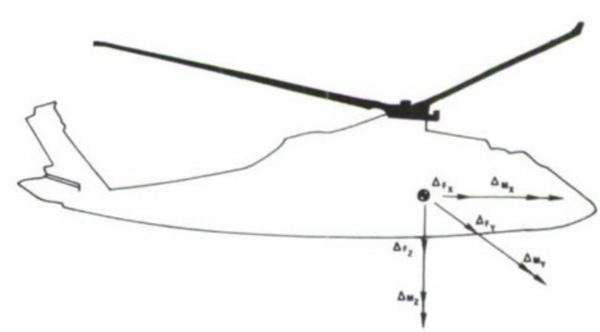


FIGURE 1 HAZEN-TYPE INCREMENTAL GENERAL FORCES AND MOMENTS ABOUT THE AIRCRAFT CENTER OF GRAVITY

STATEMENT OF PROBLEM

The helicopter dynamical flight model is highly nonlinear in comparison to a fixed-wing aircraft model. The inability to rely heavily on constant coefficient modeling techniques within the helicopter flight math model results in a complex representation of the aerodynamic characteristics of the helicopter. This math model complexity is further aggravated by the wide range of unsteady flight regimes in which the helicopter operates. This makes it necessary to model the effects of unsteady flow fields within the math model of the real-time, full-mission helicopter training simulator.

Force and moment coefficients within the helicopter math model change extensively across the flight regime of the helicopter; therefore they are best represented in tabular or data map form. Table 2 lists some of the coefficient tables used in the Black Hawk simulation. Baseline values of such coefficients are calculated by analytic prediction or come from some empirical source (e.g., wind tunnel tests). However, because of the nonlinear and unsteady effects, each flight regime of the helicopter (see Table 1) subjects the baseline coefficients to extensive modifications in order to better predict the true flight characteristics of the helicopter. These necessary modifications differentiate the training simulator math modeling task from the engineering research and development aerodynamic math modeling task.

TABLE NAME		INDEPENDENT VARIABLES
C_H	MAIN ROTOR DRAG FORCE	$(\lambda_0, \mu_0, \theta_{MR})$
C_Y	MAIN ROTOR SIDE FORCE	$(\lambda_0, \mu_0, \theta_{MR})$
C_T	MAIN ROTOR THRUST FORCE	$(\lambda_0, \mu_0, \theta_{MR})$
C_Q	MAIN ROTOR TORQUE	$(\lambda_0, \mu_0, \theta_{MR})$
$C_{T_{TR}}$	TAIL ROTOR THRUST	$(\lambda_{0_{TR}}, \mu_{0_{TR}}, \theta_{TR})$
$C_{Q_{TR}}$	TAIL ROTOR TORQUE	$(\lambda_{0_{TR}}, \mu_{0_{TR}}, \theta_{TR})$
L_{HT}	HORIZONTAL TAIL LIFT	(α_{HT})
D_{HT}	HORIZONTAL TAIL DRAG	(α_{HT})
L_{VT}	VERTICAL TAIL LIFT	(β_{VT})
D_{VT}	VERTICAL TAIL DRAG	(β_{VT})
$M_{X_{FUS}}$	FUSELAGE X-MOMENT	(β_{FUS})
$M_{Y_{FUS}}$	FUSELAGE Y-MOMENT	$(\alpha_{FUS}, \beta_{FUS})$
$M_{Z_{FUS}}$	FUSELAGE Z-MOMENT	(β_{FUS})
L_{FUS}	FUSELAGE LIFT	$(\alpha_{FUS}, \beta_{FUS})$
D_{FUS}	FUSELAGE DRAG	$(\alpha_{FUS}, \beta_{FUS})$
Y_{FUS}	FUSELAGE SIDE FORCE	$(\alpha_{FUS}, \beta_{FUS})$

α = angle of attack
β = angle of sideslip
μ_0 = advance ratio
λ_0 = inflow ratio
θ = collective pitch of the rotor blades

TABLE 2 BLACK HAWK AEROMODEL COEFFICIENT TABLE

Understandably, manual manipulation of these data tables to satisfy a static flight test data point is a tedious process at best. Automating this process as much as possible becomes a necessity.

The AUTOCOR technique is valid with any simulation model or portion of that model. The desired forces and moments to be modified need not preexist as coefficients in tabular form within the simulation model. For example, the rotor thrust determined by a blade element rotor model is analytically calculated from C_l and C_d data for the individual rotor blades. The engineer could use AUTOCOR either to modify the C_l and C_d tables or, more directly, to create a supplementary table of additional main rotor thrust values necessary to satisfy a series of static FTD points. The following discussion uses "level flight control trims" as a specific example. The techniques discussed are, of course, more widely applicable.

FORMULATION OF SOLUTION

AUTOCOR is divided into two phases or steps. Phase One allows the engineer to choose and systematically solve for coefficient table changes necessary to meet flight test data. The program outputs "delta" files containing the requested modifications in terms of increments to model coefficients. Phase Two of the AUTOCOR program incorporates these modifications back into the baseline model coefficient tables, smoothing the necessary changes in the original data set.

Phase One: Determination of Deltas

A "complete" data set for a static level flight condition consists of pilot control positions and aircraft attitudes as well as shaft horsepower, as shown in Table 3. The engineer selects the coefficients to be modified by AUTOCOR until the FTD case is matched. A typical level flight control position test set is shown in Figure 4. Table 3 lists an example of an engineering estimate of which coefficient tables might be responsible for the model's failure to match FTD for this test set. This solution is unique: the assignments could be rearranged within the table, possibly resulting in a slower convergence, but the answer would be the same. Allowing the engineer to choose which coefficient is suspect is a modification of the system described by Hazen.[1] All modeling errors were previously attributed to general forces and moments applied at the center of gravity. The predominance of main rotor force and moment contributions in the example is indicative of the main rotor's powerful influence on the entire aircraft trim condition.

FLIGHT TEST DATA VALUE: (FIGURE 2, 90 KNOTS)		"BEST GUESS" COEFFICIENT TABLE TO BE MODIFIED:	HAZEN "BEST GUESS" ASSIGNMENTS:
θ_b	PITCH ATTITUDE	$C_{H_{MR}}$	M_Y
ϕ	ROLL ATTITUDE	$C_{Y_{MR}}$	M_X
δ_a	LATERAL CYCLIC	LATERAL FLAPPING (b_1)	F_Y
δ_b	LONGITUDINAL CYCLIC	LONGITUDINAL FLAPPING (a_1)	F_X
δ_c	COLLECTIVE CONTROL	$C_{T_{MR}}$	F_Z
δ_d	DIRECTIONAL CONTROL	$C_{T_{TR}}$	M_Z
SHP	SHAFT HORSEPOWER	$C_{Q_{MR}}$	-

TABLE 3 ENGINEERING "BEST GUESS" FOR CORRELATION LOOP ASSIGNMENTS FOR BOTH CURRENT ALGORITHM AND HAZEN-TYPE ALGORITHM

FLIGHT TEST DATA VALUE: (FIGURE 2, 90 KNOTS)	CORRELATED TRIMMER RESULT:	BASELINE COEFFICIENT TABLE VALUE:	MODIFICATION TO TABLE NECESSARY TO MATCH FTD:
θ_b = -0.4688 DEGREES	0.4669	$C_{H_{MR}}$ = 0.00080	$C_{H_{MR}}$ = 0.00019
ϕ = SET TO 0.0	0.0000	$C_{Y_{MR}}$ = 0.00002	$C_{Y_{MR}}$ = 0.00000
δ_a = 51.1192 PERCENT	51.1641	b_1 = -0.55 DEGREES	b_1 = -0.48
δ_b = 41.0766 PERCENT	41.4039	a_1 = 1.69 DEGREES	a_1 = 1.72
δ_c = 46.7776 PERCENT	46.7931	$C_{T_{MR}}$ = 0.0071	$C_{T_{MR}}$ = 0.00024
δ_d = 50.8379 PERCENT	50.8361	$C_{T_{TR}}$ = 0.0100	$C_{T_{TR}}$ = -0.00234
SHP = NOT AVAILABLE	1182.04 HP	$C_{Q_{MR}}$ = 0.00034	$C_{Q_{MR}}$ = NOT CALCULATED

TABLE 4 EXAMPLE COEFFICIENT MODIFICATION FOR A 90-KNOT LEVEL FLIGHT CONTROL TRIM POINT CORRESPONDING TO THE FLIGHT TEST DATA OF FIGURE 4

For the Black Hawk, original main rotor and tilt rotor coefficient data were computed off line, in non-real time, by the airframe manufacturer's preferred rotor analysis models. Thus the baseline table for static conditions represents the most accurate analytic model available.

Phase One is constructed of trimmer and correlator programs, usually running simultaneously.

Trimmer. The trimmer program is a trimming algorithm that iterates aircraft orientations and pilot control positions to obtain a zero-acceleration condition (much the same as a pilot in the loop would). It must also maintain specified airspeed, altitude, and sideslip angle for level flight. Trimming algorithms are available for all of the "standard" types of flight conditions corresponding to the flight tests listed in Table 1. The FTD correlation is successful when the resulting pilot control positions and attitudes computed by the trimmer match those of the FTD.

Correlator. The correlator is an extension of the aircraft trimmer concept, since they both drive the accelerations to zero. However, pilot control positions and aircraft attitude are also specified. The modeled vehicle forces and moments are modified (through the specified coefficients) to provide the desired unique solution shown as an example in Table 4. In order to run the isolated (i.e., no trimmer) correlation algorithm, the engineer must either have a full set of FTD — θ_b, ϕ, δ_a, δ_b, δ_c, δ_d, Shaft Horsepower — or a "best guess" for any missing data.

Problem of Missing Data. Usually, FTD reports do not contain complete data sets for most test cases. In Figure 4, for example, roll attitude and shaft horsepower are missing. Occasionally, incomplete individual FTD cases can be combined to create a more complete data set. This is the case, most often, for control trims and level flight performance. A framework across gross weight and altitude, using these instances of relatively complete data points, provides the engineer with a gauge to judge the accuracy of coefficient table modifications made by less completely defined points.

The second, more frequently used approach is to estimate values of the missing data. This requires extrapolation from similar but more complete data, a best guess, or the advice of an experienced test pilot. Another method of supplying the missing data is to derive these data by the trimmer algorithm. This method, in the case of very sparse data (e.g., rate of climb as a function of shaft horsepower and airspeed), can create apparent data discrepancies in the maps. This happens when the trimmer provides control positions and aircraft attitudes significantly different from those of the aircraft. In effect it creates a bad data point for the correlator to match.

Once the correlator has converged on a stable solution, a record of the trim is made. This record contains the values of the independent variables for each modified coefficient. Also contained in this record is the baseline coefficient (the value interpolated from the original table) and the increment the correlator was forced to add to trim the aircraft. This record is gathered with similar records from other tests to form a "delta" file. This file forms the principal input to Phase Two of the AUTOCOR program.

Phase Two: Incorporation of the Function Modifications

Phase Two of the AUTOCOR program is responsible for modifying the baseline coefficient tables using the "delta" file from the correlator. Three numerical techniques form the backbone of this program. The remainder of the program deals with file input/output, plotting, and various other ancillary functions.

Before discussing implementation of AUTOCOR, a brief survey of the mathematical problem of incorporating deltas back into the baseline tables is helpful. As shown in Figure 2, the most

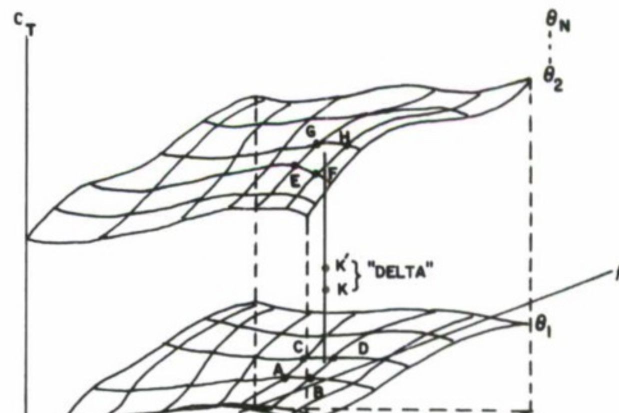


FIGURE 2 TWO SURFACES OF THE TRIVARIATE FUNCTION C_T (λ_o, μ_o, θ)

general case for the Black Hawk simulator math model is a function of three variables. For example, C_T of the main rotor is represented by "stacked sheets," each of which is a lattice of discrete, unequally spaced function values at particular values of the independent variables λ_0, μ_0, and θ. The functional value indicated by point "K" can be obtained by linearly interpolating among the immediately surrounding points "A" through "H." The Phase One correlator determines a "delta" such that the new required value is "K'." The problem is then how to modify points "A" through "H" so that a linear interpolation between the new points will yield "K'." The problem is complicated by the fact that more data points in addition to "K'" may be within the region defined by "A" through "H." There are also many adjacent regions making up the total functional envelope defined in Figure 2 which may also contain deltas. Phase Two of AUTOCOR addresses these problems. An overview of its numerical techniques and implementation follows.

Numerical Techniques for "Delta" Incorporation. The function tables are discrete, with unequal spacing of independent variable coordinates, or "breakpoints." The three techniques currently in use are simple averaging, slope ratio averaging, and linear regression. The first method, simple averaging, takes an arithmetic average of all the deltas that affect a given breakpoint. In Figure 3, the four deltas x_4, x_5, x_6, and x_7 all affect the modification of the coefficient at the breakpoint FUS_7. The average of the four points is applied to the breakpoint. This method is then applied to each breakpoint that has a delta (or deltas) adjacent to it.

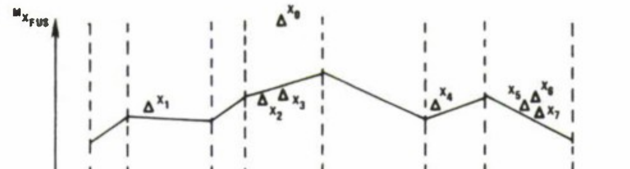


FIGURE 3 TYPICAL MONOVARIATE FUNCTION

The slope ratio averaging technique is quite similar except that a weighting feature is included. The amount a delta can affect a breakpoint is proportional to its proximity to that breakpoint. A delta very close to a breakpoint would be weighted more heavily than a delta farther away (such as x_4 in relation to βFUS_7).

The last method uses linear regression to modify the table. A least-squares criterion minimizes residual errors when fitting a line to the original table points and the delta points. The regression line is passed through the data table along each of its axes. In the case of a trivariate table, three regression lines are drawn through each breakpoint. The average of the three is used as the final value at that breakpoint. Flight test data tends to create isolated clumps of deltas, regions of the coefficient maps where many test points fall, separated by large areas of no experimental data. Frequently, several deltas fall between adjacent breakpoints as a result of this clumping of data. The engineer must choose a numerical method best suited to the available data.

Each of these techniques has its limitations and strengths. Simple averaging is very useful if there is a low level of confidence in the original baseline table. Simple averaging can, in one pass, restructure a table. This technique would be useful for the initial creation of a table. For example, since lateral flapping is calculated by an analytic expression in the Black Hawk math model, a coefficient table was created to house the delta corrections in lateral flapping necessary to fine-tune lateral control position. Its strengths are also its limitations. It is poor for fine-tuning a table because averaging can change the breakpoint values by large amounts in a single pass.

Slope ratio averaging is useful for restructuring a new map while using the delta information to improve the slopes. However, like simple averaging, it is not useful for fine-tuning the maps.

Linear regression is the preferred method during the later stages of map manipulation. It maintains a more continuous slope and filters out large changes. For this reason it may take several correlator passes to match FTD.

Curve Smoothing Techniques. Frequently delta incorporation by the methods just described yields "spikes" or rough spots which can lead to dynamic oscillations as the model passes through static solutions. In Figure 3, x_0 represents a delta point which would likely create a spike in the otherwise smooth function curve. To remedy this problem (which is caused by insufficient and inconsistent data) a curve smoothing technique is desirable. AUTOCOR provides three curve fitting methods: polynominal curve fit smoothing, linear regression smoothing, and modified data smoothing.

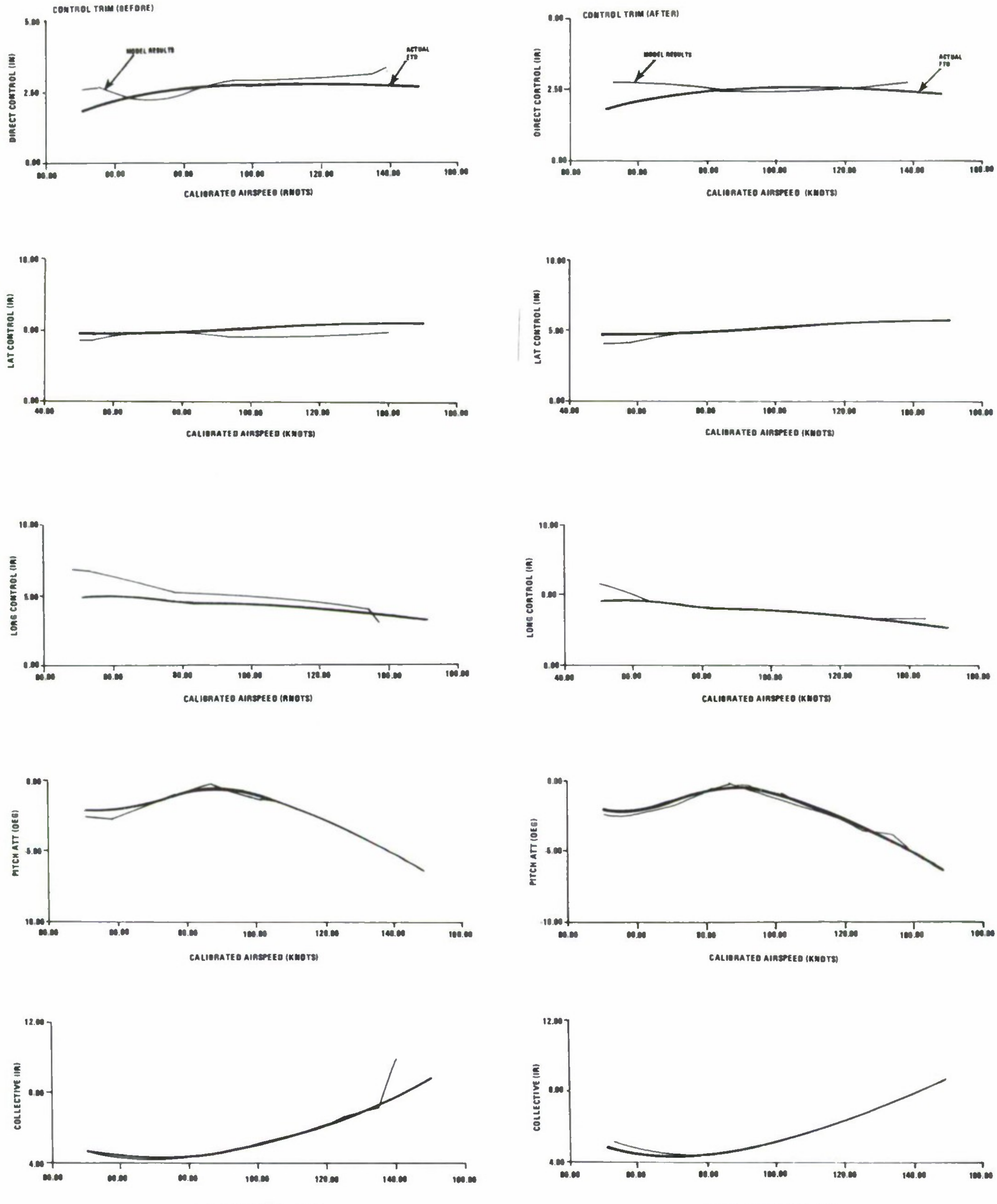
The polynomial fit smoothes the entire data table. Thus a single "spike" may have a powerful influence on the entire map. This method is seldom used because the resulting curve shapes frequently depart too far from the baseline data.

The linear regression method is more localized about the region of the "spike." It is the preferred method for repeated overall smoothing of the maps because it gradually removes spikes without drastically distorting overall map shape. This method fits a line based on a least-squares criterion through successive sets of three breakpoint function values and then averages the computed values at each breakpoint.

The modified data smoothing method allows the number of adjacent breakpoints included in each regression to be specified. This method also has the characteristic that only breakpoints modified by deltas are affected. This allows, for example, linear regression on cruise condition points without changing a satisfactory slow speed regime.

The improvement in model fidelity possible after a single pass through AUTOCOR is shown in Figure 4. In this example full linear regression smoothing of the maps was used to incorporate the delta points. Repeated passes through the AUTOCOR algorithm would further improve data matching in the low-speed regime.

Implementation of AUTOCOR Task. The user begins by loading the numerical values of the baseline data tables from the simulator model into AUTOCOR. Both function and independent variable values can be readily obtained from these files. The user then runs the correlation. The user has sub-options to correlate one or all of the coefficient functions and, after execution, can rerun the analysis using a new (or the same) function as well as a new (or the same) method. The option to smooth the computed function curves to reduce the effects of peaks resulting from erroneous data is always available. During or after cycles of correlation and smoothing, the user may choose 2-D or 3-D (perspective) plots of any function. Cross plots for independent variables are also available. AUTOCOR outputs the tables in a form immediately usable by the simulator models.



(a) BEFORE

(b) AFTER

FIGURE 4 MODEL RESULTS BEFORE AND AFTER AUTOCOR ADJUSTMENT

RESULTS AND CONCLUSIONS

The AUTOCOR program represents a first attempt to fully automate the flight test data correlation task within the development of a training simulator. Table 4 is an example of the typically excellent performance of the AUTOCOR algorithm on an isolated flight test data point. This level of accuracy can usually be maintained within a flight test sweep (Figure 4). However, combining individual test sweeps (i.e., over the airspeed range +50 to +150 knots) creates a more challenging task. Major conclusions regarding the performance of the AUTOCOR techniques, as demonstrated during the Black Hawk project, are:

1) Current application of AUTOCOR is limited in part by the type of flight test data available. Flight test data must be self-consistent and complete or conflicts will arise within the coefficient maps.

2) As a result of (1), AUTOCOR is best applied in a series of small steps (for example, one type of flight test at a time, or even one test case at a time) so that only small regions of the maps are altered. Conflicts between correlation runs must be carefully studied and resolved.

3) The most efficient uses of AUTOCOR techniques are methods which allow the engineer to be interactive with the process, particularly with respect to selecting automated options and monitoring results between AUTOCOR steps. This must be done in order to select the best option for any succeeding step in the process.

4) The AUTOCOR concept was applied for the first time to a helicopter training simulator development project (Black Hawk). The conclusion from this initial application is that further development of AUTOCOR can yield an effective, powerful design tool for future use.

RECOMMENDATIONS

1) Pilot-in-the-loop testing of the simulator must be performed parallel to the FTD correlation effort. Modifications to the coefficient tables must be evaluated for their effect on dynamic flight characteristics.

2) Helicopter training simulators are often developed years after the aircraft airworthiness flight test program has been conducted. In the future the simulator and aircraft manufacturers should work together in preparing a flight test program that will supply the needs of the simulator development program as well as the aircraft development program.

3) Methods must be developed to more readily identify inconsistent flight test data and assist the engineer in reconciling pilot opinion with FTD.

4) Expanded wind tunnel tests for aerodynamic forces and moments of aircraft components and their interactions would greatly enhance the accuracy of the baseline model. High-angle-of-attack, rearward, and sideward flight regimes often lack fuselage and empennage components.

REFERENCES

1. Hazen, M. R., "Use of Flight Test Results to Improve the Flight Simulation Fidelity of the LAMPS MK III Helicopter Operational Flight Trainer," presented at the I/ITEC conference, Washington, D.C., October, 1984.

ACKNOWLEDGEMENTS

The authors wish to acknowledge the assistance of Stanley Briczinski (Staff Scientist), Skye Sweeney, and Tom Stockdale, all currently with Singer-Link, during development of the AUTOCOR program.

ABOUT THE AUTHORS

Susan C. Garing is a systems engineer with Singer-Link Flight Simulation. She is currently involved with the UH-60 Black Hawk full mission simulator program being conducted for the U.S. Army by Singer-Link. She has a BS from Princeton University and an MS from Rensselaer Polytechnic Institute in Aeronautical Engineering and Helicopter Technology.

Michael L. Brychcy is a systems engineer with Singer-Link Flight Simulation. He is currently working on flight software design and integration for the Qantas Boeing B747-438 commercial simulator. He holds BS and MS degrees in Aerospace Engineering from the State University of New York at Buffalo.

AN EXPERIMENTAL ANALYSIS OF CRITICAL VISUAL DISPLAY PARAMETERS FOR COMPUTER-BASED TRAINING AND JOB PERFORMANCE AIDING

Rosemary Dunlop Garris
Daniel J. Dwyer
Patrick J. Moskal*
Christopher P. Mulligan*
Naval Training Systems Center
Orlando, Florida

ABSTRACT

The growing reliance on video display units to present graphic information in support of both military training and job aiding, is expected to continue. Empirical research has provided guidelines for display parameters associated with alphanumeric (textual) information, however research concerning graphics (particularly line drawings) is limited. This paper discusses the results of recent experiments which explored the effect of critical visual display parameters on task performance using line drawings as stimulus materials. The results suggested that in many cases, very low levels of graphics detail may be sufficient to produce adequate response times in locator task performance. Additionally, it is noted that, production of graphics with low levels of detail result in dramatic cost savings.

INTRODUCTION

The proliferation of training systems throughout the military services has contributed to a shifting trend, away from paper-based presentation media associated with traditional classroom lecture to computer-driven presentation formats. Training systems, as well as automated job performance aids (JPAs), rely heavily on visual presentation of information to support training and task performance. Consequently, the visual display characteristics of these systems have become a critical component in the user-system interface. Poor design, inaccurately specified visual display parameters, and/or omission of critical interface components associated with the delivery media can hinder legibility, resulting in systems which are not used or which may prove to be ineffective in providing technical information necessary for training and task performance.

Training systems often incorporate display images of training relevant equipment (e.g., via a videodisc system) with supplemental graphic line drawings, computer-generated graphic overlays, and textual information. Similarly, JPAs, which have traditionally existed in hardcopy (paper) formats and have now transitioned to computer-based modes, provide electronic delivery of technical information (schematics, illustrated parts breakdowns, etc.) through microprocessor control. This adaptation of hardcopy graphics to automated display media, makes it crucial to optimize graphics production efficiency and to determine the effects of critical visual display parameters on task performance and training effectiveness.

Past research has provided guidelines on display parameters associated with alphanumeric (textual) information. Meister (1984) provides a thorough review of this research, documenting efforts associated with display parameters such as symbol size, character fonts, symbol height-to-width ratios, and so on. However, empirically-based guidelines for graphics (especially line drawings) are lacking (Sweezey and Davis, 1983).

Recent research (Dwyer, 1985), which focused on locator task performance using a CRT display depicting printed circuit boards, revealed that a small (5" X 5") display screen resulted in acceptable locator task performance (measured by accuracy rates), but only when high discriminability existed among the components which made up the graphic. When low discriminability (a densely packed, cluttered display) existed in the graphic, a larger display screen (9" X 9" or 12" X 12") was warranted.

While this finding may not be a critical factor for school-based training systems, it is a critical issue in the development of portable, lightweight JPAs since screen size has a direct bearing on device size and weight characteristics. Thus, some of the preliminary research suggests that optimal levels of critical visual display parameters may vary as a function of intended application.

The Naval Training Systems Center (NAVTRASYSCEN) is currently engaged in a research program to identify optimum levels of critical visual display parameters within different domains (e.g, training, job aiding). The purpose of this paper is to present the results of two recent experiments in this area which explored the effects of certain display parameters on task performance.

EXPERIMENT 1

Despite the findings of Dwyer (1985) which advocated larger screen sizes for some tasks, portability remains a critical design issue in the development of automated JPAs. As a result, alternative techniques must be sought which enhance the legibility of graphics on small display screens. Experiment 1 was based upon the work of Regal and Knapp (1984), who found improvements in performance accuracy when unnecessary information was deleted from a visual display. One intent of this joint Navy-Air Force research was to remove non-critical areas of the graphics display in order to produce less clutter and facilitate task performance. The purpose of the experiment was to determine if reduced levels of graphics detail could compensate for performance decrements associated with small screen clutter. The

*This research was supported in part by Patrick J. Moskal's and Christopher P. Mulligan's appointment to the research participation program at the Naval Training Systems Center (NTSC), administered by Oak Ridge Associated Universities through an agreement between the U. S. Department of Energy and the NTSC.

experiment examined the effects of screen size (7" X 7" and 12" X 12"), image resolution (35 and 280 pixels per inch), and level of detail (high, medium, and low) on disassembly/assembly task performance. The levels of resolution were selected based on commercial availability and represented the extremes, such that any resulting performance differences would be exaggerated. High level of detail was operationally defined as all (100%) of the detail on the actual equipment used in the experiment; medium detail was defined as the component to be removed/installed, all immediate surrounding components, and the outline of the bomb ejector rack; and low detail was defined as the component to be removed/installed and the outline of the bomb ejector rack.

METHOD

SUBJECTS

Sixty Air Force maintenance training students from Lowry AFB, CO served as participants in the experiment. The students ranged in age from 17 to 31 years (mean = 20.0 years), and in tenure from 1.5 to 48 months (mean = 5.4 months).

PROCEDURE

All subjects were seated (individually) in front of a Megatek 7210 high resolution monitor which presented graphic displays of an Air Force bomb ejector rack (model MAU-12B/A). Twelve disassembly and 12 assembly frames were presented, each of which contained a graphic display of the bomb ejector rack and textual instructions explaining how to perform the task. An actual bomb ejector rack and the tools necessary to perform each disassembly/assembly task step were located on a workbench placed between the student and the monitor.

The experimental design was a 2 X 2 X 3 between subjects factorial with 5 (randomly assigned) subjects in each of the 12 cells. Performance measures were response accuracy (number of correct tasks performed), absorption time (time to read and interpret the frame), and manipulation time (time to perform the task following absorption time). Response accuracy was assessed by the experimenter and entered into the computer for storage and subsequent data analysis. Absorption and manipulation times were recorded by the computer, based upon experimenter input (i.e., a "stop" command was entered when the student had stopped reading (absorbing) and began manipulating).

RESULTS AND DISCUSSION

A multivariate analysis of variance (MANOVA) was used to analyze the results due to the multiple dependent variables assessed. The results of the analysis revealed no significant main effects nor any significant interactive effects, $p > .05$ in all cases. Although these results suggest that neither screen size, image resolution, nor level of detail significantly affected task performance, these results must be treated with extreme caution due to the small sample size (5 per cell) and the subsequent loss of statistical power to detect true differences.

EXPERIMENT 2

Independent variables such as screen size and image resolution are relatively easy to quantify because they can be defined and measured in discreet increments. Consequently, it is a straightforward process to select "levels" of these variables for examination. For example, virtually any level of screen size (5" X 5", 7" X 9", 12" X 15", etc.) can be selected for study. Similarly, image resolution can be defined and measured by the number of pixels per inch, hence we can select and study a particular level of resolution of interest (provided the display media can accommodate the desired level). However, measuring variables such as graphics detail is not so easy. It was therefore necessary to systematically develop a method for quantifying level of graphics detail, such that further experimentation could proceed using a standard metric with generalized applicability. Several techniques are available for operationally defining graphics detail. Four are presented below:

1. Method of cue presentation: Cues can be added to a graphic either (a) concentrically, (b) randomly, or (c) as a function of unique/outstanding landmarks.

2. Number of cues added: The number of cues can be increased some pre-determined number at a time, or by some percentage of the total number of cues on the actual equipment.

3. Amount of target detail: The amount of detail inherent in the target component can be varied (such as the appearance of pointers on dials, tick marks, etc.).

4. Amount of cue detail: The amount of detail inherent in the cue components can be varied (similarly to that of the target detail).

The scope of this experiment was limited to the first 2 techniques addressed above, method of cue presentation and number of cues added. (Examples of which can be found in Figures 1, 2, and 3.) In order to reduce the number of cue components which must be generated to a manageable number (such that graphics production is more efficient), a pilot study was run. Pilot data were gathered to determine the level of detail (number of cues added) at which locator task performance began to stabilize. These data identified the point at which added detail failed to produce measurable gains in locator task performance. Level of detail in paper-based line drawings of a printed circuit board and an Air Force bomb ejector rack (See Figure 4 for full detail illustrations of the two pieces of equipment.) was systematically increased by adding cue components, one-at-a-time, concentrically surrounding the target components, until subjects correctly identified the correct target component. Based upon the data collected, a range of detail levels was established for the subsequent locator tasks and was set at 1, 3, and 5 cue components for the bomb ejector rack, and 2, 6, and 10 cue components for the printed circuit board. The pilot data suggested that the overwhelming majority of students were successful in locating target components within these bounds.

Figure 1. Concentric Cue Addition

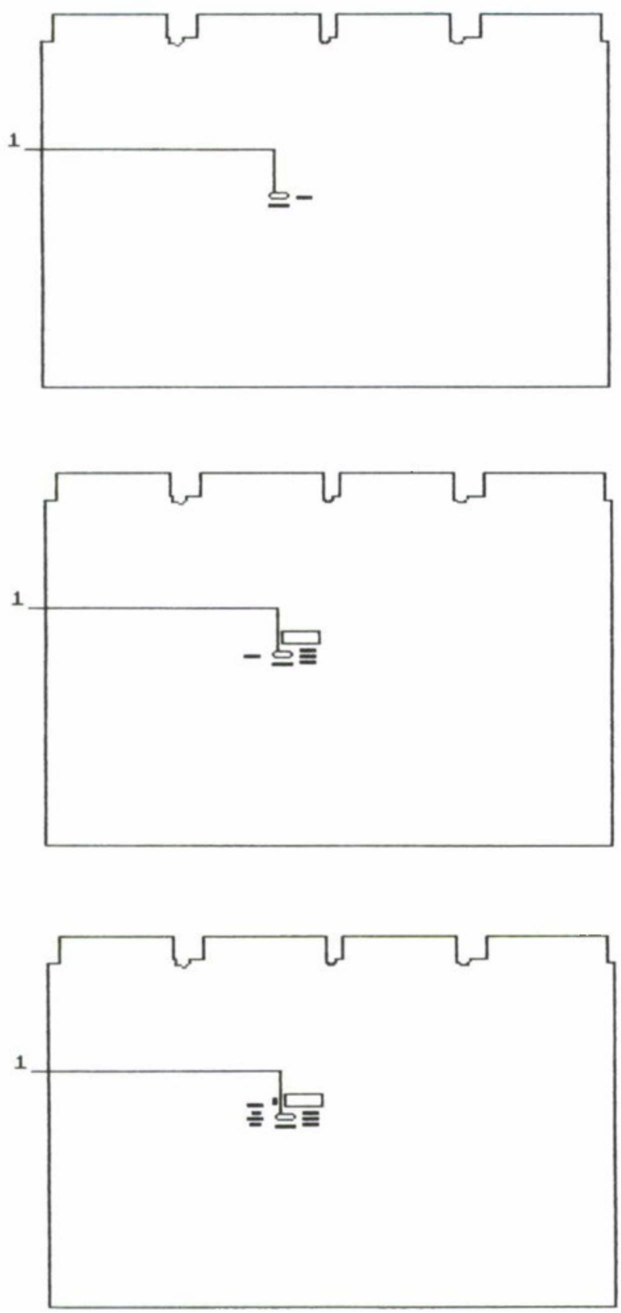


Figure 2. Random Cue Addition

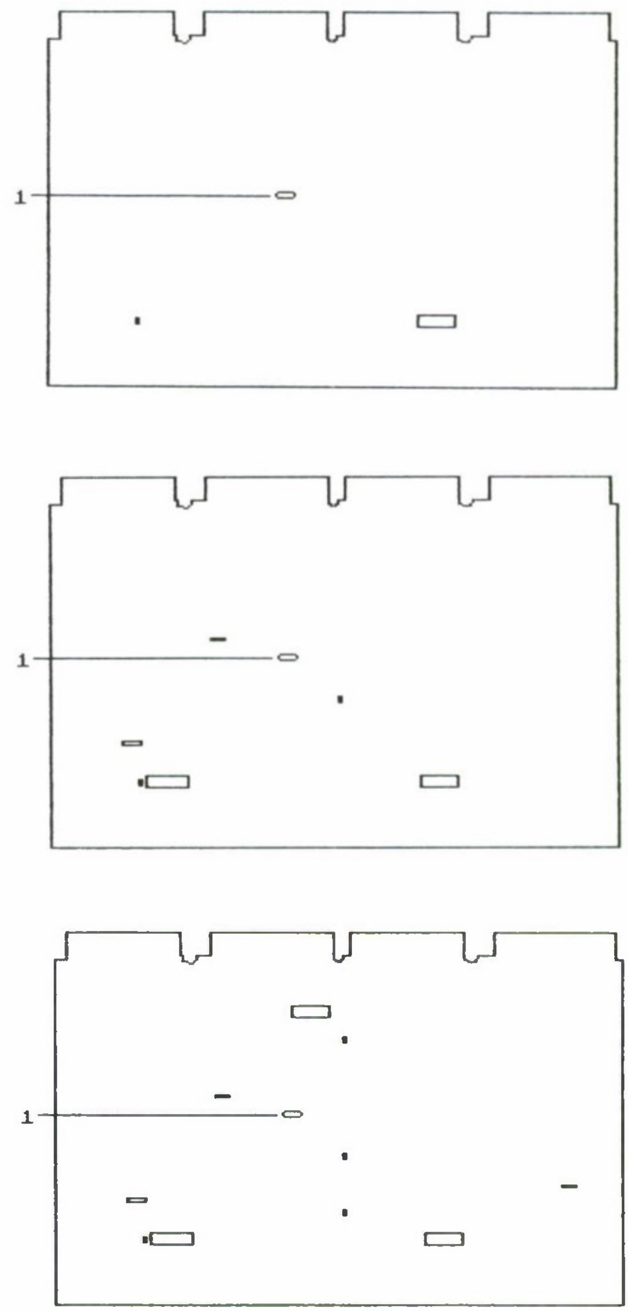


Figure 3. Landmark Cue Addition

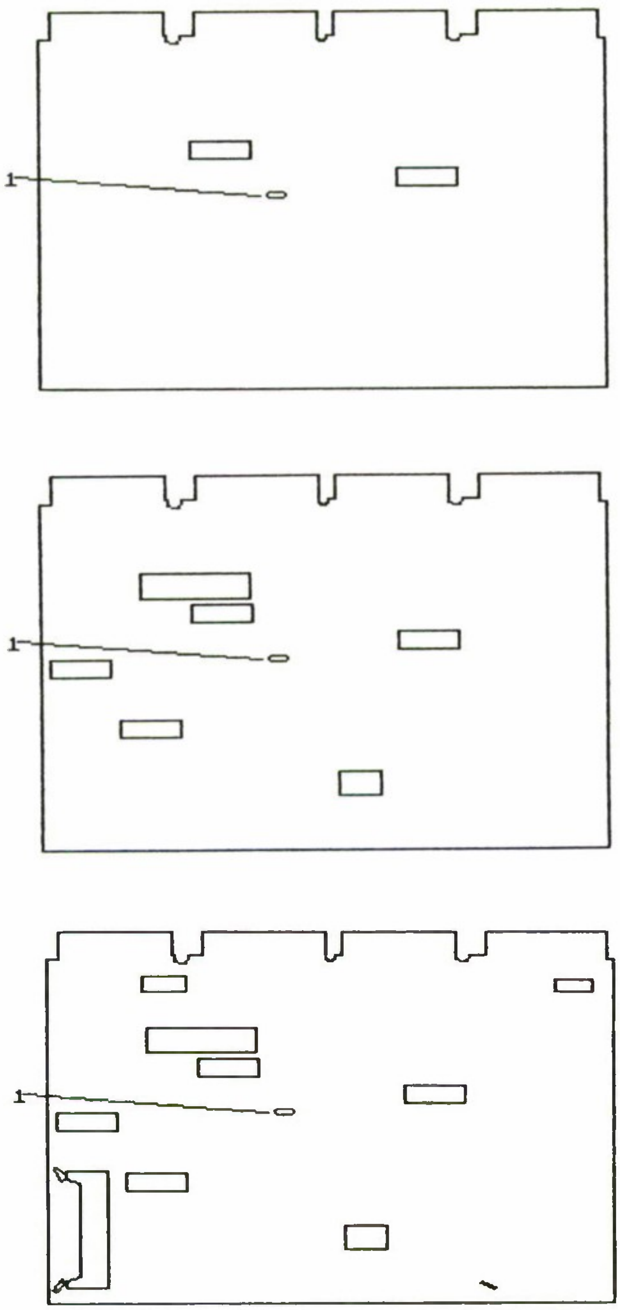
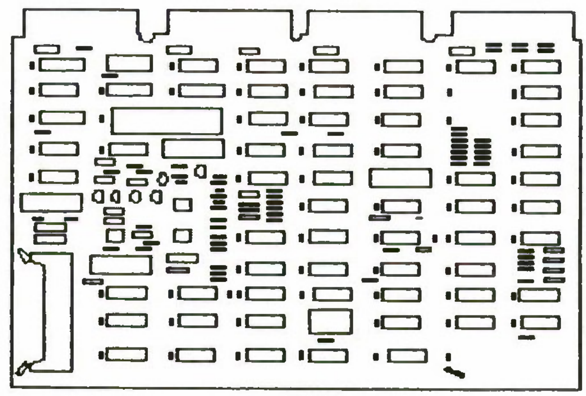
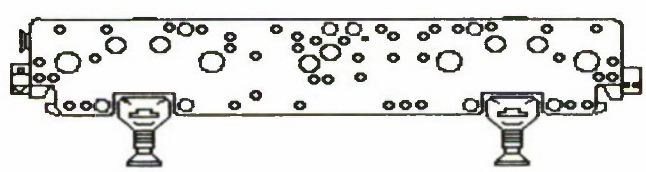


Figure 4. Circuit Board and Bomb Ejector Rack with All Components



Circuit Board



Bomb Ejector Rack

METHOD

SUBJECTS

Eighty-five male students from Torpedoman's "A" school and 25 male students from Nuclear Power "A" school participated in the experiment. All 110 students were from Service School Command, Naval Training Center, Orlando, Florida.

PROCEDURE

The three levels of detail addressed above and three methods of cue presentation (concentric, random, and landmark) were manipulated in a locator task procedure involving two equipment test beds: a printed circuit board, and an Air Force bomb ejector rack. Two additional conditions, a verbal description (of the target component and cues) and a target-only with the equipment outline, served as controls. Nine fifteen-page stimulus groups of line drawings were used for each test bed, one group corresponding to each of the experimental conditions. The same fifteen target components, albeit at different levels of the two independent variables, were depicted in the stimulus groups for the nine experimental graphics cells and in the two control conditions.

Subjects were randomly assigned to one of the 11 conditions. The order in which each subject performed the locator tasks was counterbalanced across test beds in order to control for practice effects. Additionally, the fifteen pages, each containing a different target component, were randomly presented to the student. All students performed the locator task for both pieces of equipment.

Subjects were seated at a table which held either the printed circuit board or the bomb ejector rack and the corresponding set of stimulus materials. The experimenter presented one drawing at a time and asked the student to locate and identify the target component on the actual printed circuit board/bomb ejector rack that was identified by a callout (line which pointed to the target) in the line drawing. The experimenter recorded accuracy (correct/incorrect) and task time using a stop watch. Time was measured from the point when the drawing (or verbal description) was first presented until the student correctly identified the target component or when he indicated that he could not locate it. Prior to the actual data collection, students were given two practice trials (with a target not used for actual data collection).

RESULTS

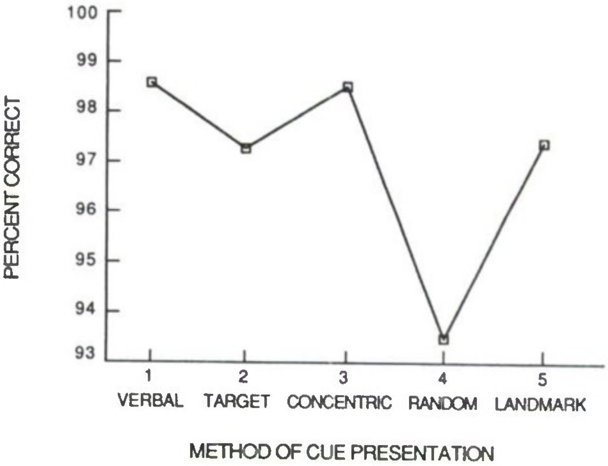
For each subject, fifteen time scores (in seconds) were recorded, as well as an accuracy score reflecting the percentage of target components which were identified correctly. A mean for each subject was then computed for the fifteen times scores. This score and the percent correct represent the two dependent variables, time and accuracy respectively. The findings of the data analysis are presented below.

ACCURACY

Printed Circuit Board. A one-way ANOVA was computed on the printed circuit board accuracy scores for method of cue presentation in order to assess the verbal only and target with outline methods with the other methods of cue presentation. This analysis yielded a significant main effect, $F=3.51(4,105)$, $p < .01$, (See Figure 5 for the graphed means). A Scheffe post hoc test identified a significantly greater level of accuracy for concentric presentation of cues over that of the random presentation of cues, ($p < .05$). None of the other pairwise comparisons were statistically significant. Results of a one-way ANOVA on level of detail indicated no significant effects in accuracy rates across the detail levels.

Next, a two-way ANOVA using accuracy scores was computed, with method of cue presentation and level of detail as the independent variables (excluding verbal and target only conditions). A main effect for method of cue presentation was found which supports the results of the one-way ANOVA which examined this variable, ($F(2,81)=5.96$, $p < .01$). Again, the findings suggest level of detail does not significantly affect task performance. The method of cue presentation was an important factor, however, with the concentric method found to produce greater accuracy than the random method.

Figure 5. Circuit Board - Accuracy All Conditions

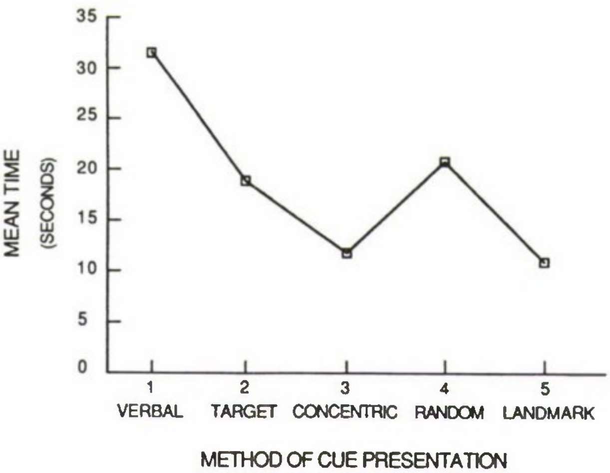


Bomb Ejector Rack. Because a significant heterogeneity of variance was present, a Kruskal-Wallis one-way ANOVA was computed on the method of cue presentation. This analysis yielded no significant effects. This was also true for a one-way ANOVA computed for level of detail. Next, a two-way ANOVA using accuracy scores was computed and this too, yielded no significant effects. Apparently, neither method of cue presentation nor level of graphic detail affects location performance accuracy on the bomb ejector rack.

TIME

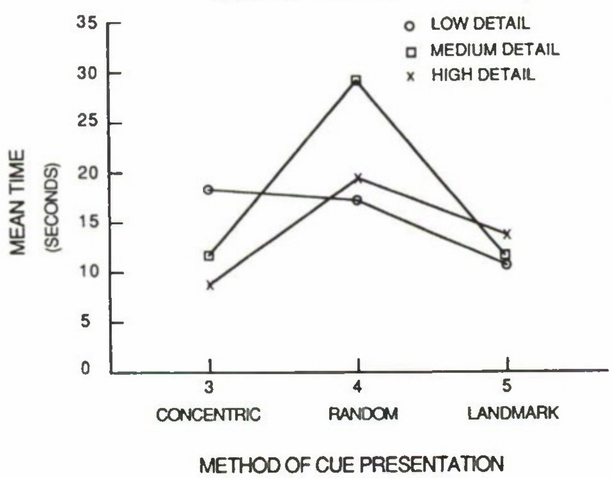
Printed Circuit Board. In order to determine if method of cue presentation affected the task time, a Kruskal-Wallis one-way ANOVA was computed for methods of cue presentation, including verbal description only and target with outline only. This analysis, which was used because there was significant heterogeneity of variance, resulted in a significant effect, $H=25.1$, $p < .001$ (see Figure 6 for the graphed means). A Scheffe post hoc test revealed that presentation of landmark cues resulted in significantly faster location times than both the random method of presenting cues and the verbal description only method ($p < .05$).

Figure 6. Circuit Board - Time
All Conditions



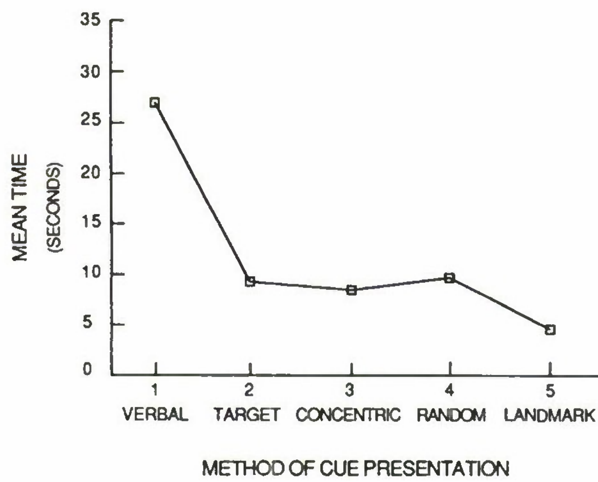
A two-way ANOVA using mean time scores for the printed circuit board was computed with method of cue presentation and level of detail as independent variables. Scores for the verbal description only and target with outline only were not included in this analysis since they were addressed previously. An interaction effect between the two independent variables was found, $F(4,81)=2.71$, $p < .05$ (see Figure 7 for the graphed means). Post hoc follow-up tests identified significant time differences between the concentric - high detail condition (mean=8.65 seconds) and the concentric - low detail condition (mean=17.91 seconds), $p < .05$, and also between the concentric - high detail condition (mean=8.65 seconds), and the random - high detail condition (mean=19.05 seconds), $p < .05$. The analysis also yielded a main effect for the method of cue presentation, $F(2,81)=8.44$, $p < .001$, confirming the results of the Kruskal-Wallis ANOVA. Finally, a one-way ANOVA performed on four levels of graphics detail (target only, two cues added, six cues added, and ten cues added) revealed no significant effects ($p > .05$) suggesting statistically equivalent location times across all levels of detail. These analyses suggest that location cues presented using the landmark and concentric methods produce faster identification of target components than either the verbal or random methods. There were no clear trends regarding the effect of level of graphics detail on task times.

Figure 7. Circuit Board - Time
Experimental Conditions Only



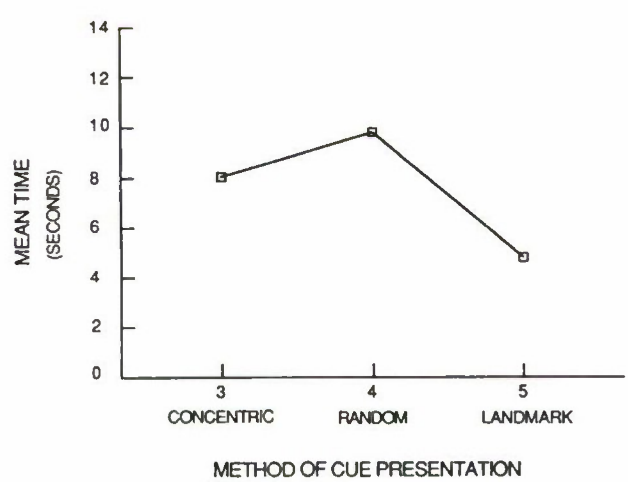
__Bomb Ejector Rack.__ Because heterogeneity of variance was evident, a Kruskal-Wallis one-way ANOVA was computed for method of cue presentation. This analysis resulted in a significant main effect, $H=29.47$, $p < .001$, (see Figure 8 for the graphed means). A Scheffe post hoc test revealed that the verbal description only method resulted in significantly slower location times that that of all of the other methods of cue presentation. A one-way ANOVA for level of detail resulted in non-significant effects for location times on the bomb ejector rack test bed.

Figure 8. Bomb Ejector Rack - Time
All Conditions



A two-way ANOVA was then computed using the bomb ejector rack mean time scores, resulting in only a main effect for method of cue presentation, $F(2,81)=4.24$, $p < .05$, (see Figure 9 for the graphed means) thereby substantiating the results of the Kruskal-Wallis ANOVA. These results, for the bomb ejector rack, support the findings obtained for performance times on the printed circuit board, such that verbal description only was found to be a poorer method of supporting a locator task than graphic methods, however, varying the level of graphic detail did not significantly affect task times.

Figure 9. Bomb Ejector Rack - Time
Experimental Conditions Only



CONCLUSIONS

ACCURACY

The findings with respect to response accuracy are equivocal. The only significant effect emanating from the results of the accuracy data analysis was that the concentric method of cue presentation resulted in higher performance accuracy rates than did the random method of presentation. This finding only held for the printed circuit board test bed. The lack of any meaningful pattern in the accuracy data might be explained by examining the method used to gather performance data.

Students attempted to locate target components on the actual printed circuit board/bomb ejector rack based upon information in the line drawing. The student continued with a task until he accurately identified the correct target. Consequently, he may have incorrectly identified several components before accurately identifying the target (correct) component. However, his response was recorded as correct, regardless of his number of "misses", as long as he ultimately identified the correct target. A task was "graded" as incorrect only if he gave up.

TIME

The results of the response time data provide only one clear conclusion - the method of verbal description consistently resulted in slower identification times. This was true for both test bed applications. These results are intuitively logical - trying to identify a physical component with only verbal instructions is a difficult task.

One interesting observation was that very low levels of detail (i.e., the target with test bed outline) resulted in response times statistically equivalent to the higher levels of detail graphics. Apparently, simply providing the outline of a piece of equipment provides a sufficient amount of information to locate targets in a timely fashion. If this finding is born out in subsequent research, it could represent significant cost savings to the military in the preparation of instructional and performance aiding graphic illustrations. For example, recently prepared job performance aids for the fire control system of the M1-Tank contain 500 graphic illustrations. If the highest possible level of graphics detail (100% of the detail appearing on the actual equipment) was required to support the maintenance of this system, it would take 37.5 weeks to reproduce these graphics on a computer display, and the approximate cost would be $67,500. However, if the lowest level of detail (target with outline of equipment only) was sufficient, it would take only 2.1 weeks to produce the necessary graphics and would cost only $3,750. When this is generalized over the massive number of equipment systems operated by the military, the magnitude of possible savings becomes apparent. (See Figures 10 and 11 to see the time and cost savings, respectively.)

Clearly, additional research in this area is warranted. Performance measures associated with location accuracy should include an assessment of the number of errors made during task performance. Also, the locator tasks were performed using paper media; it is not clear if the findings observed in this experiment will generalize to a computer-based video display unit. However, the experiments which will follow the research reported here will also be examining the dimension of computer display screen size as it relates to the level of graphics detail.

Figure 10. Production Time Differences by Level of Detail

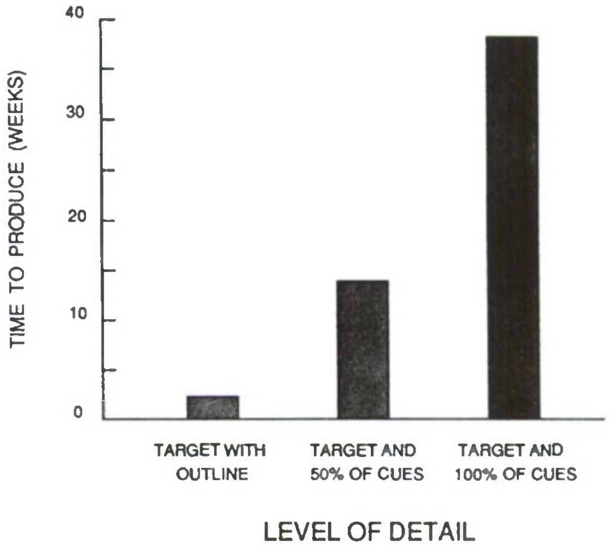
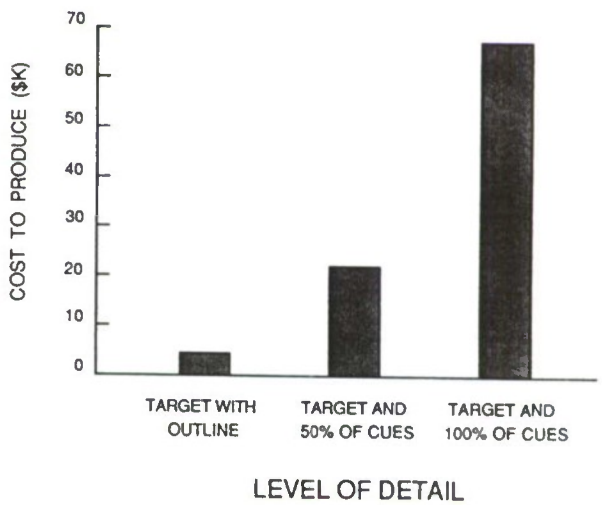


Figure 11. Production Cost Differences by Level of Detail



REFERENCES

Dwyer, D. J. (1985). An Exploratory Study of the Effect of Screen Size and Resolution on the Legibility of Graphics in Automated Job Performance Aids. NAVTRAEQUIPCEN IH-355, Naval Training Equipment Center, Orlando, FL.

Meister, D. (1984). Human Engineering Data Base for Design and Selection of Cathode Ray Tube and Other Display Systems. NAVPERSRANDCEN TR-84-51, Navy Personnel Research and Development Center, San Diego, CA.

Regal, D. M. & Knapp, B. G. (1984). An aid for improved information processing of high density computer generated visual displays, Proceedings of the Human Factors Society 28th Annual Meeting

Sweezey, R. W. & Davis, E. G. (1983). A case study of human factors guidelines in computer graphics, IEEE, Computer Graphics and Applications, pp. 21-30.

ABOUT THE AUTHORS

MS. ROSEMARY DUNLOP GARRIS is a Research Psychologist in the Human Factors Division of the Naval Training Systems Center. She has five years experience in military training research and development. Specific work experience has been in the following areas: computer-based electronic troubleshooting training; computer-based maintenance job performance aiding and training; radio instruments navigation training; chemical, biological, and radiological defense training; and visual display parameters for training and aiding applications. Ms. Garris received her B.A. from Stetson University in 1979 and her M.S. degree in Industrial/Organizational Psychology from the University of Central Florida in 1986.

MR. DANIEL J. DWYER is a Research Psychologist in the Human Factors Division of the Naval Training Systems Center. He has over eight years experience in military training R&D. Specific work experience has been in the areas of human factors front-end analysis for maintenance training, development of design guidelines for maintenance simulators, evaluation of visual display parameters for training and aiding applications, computer-assisted instruction, and embedded training. Mr. Dwyer received his B.S. degree from SUNY Cortland in 1977 and his M.S. degree in Industrial/Organizational Psychology from the University of Central Florida in 1979. Currently, he is working on his Ph.D. in Industrial/Organizational Psychology at the University of South Florida.

DR. PATRICK J. MOSKAL is a Post-doctoral Fellow appointed to the research participation program in the Human Factors Division of the Naval Training Systems Center (NTSC) administered by Oak Ridge Associated Universities through an agreement between the U.S. Department of Energy and the NTSC. His appointment is in its second year. Dr. Moskal's research has focused on performance under stress, development of training aids for chemical, biological, and radiological defense, and the development of micro-computer based part-task training. He received his M.A. and Ph.D. in Experimental Psychology from the University of Notre Dame, specializing in Sensation and Perception and Psychophysical Scaling.

MR. CHRISTOPHER P. MULLIGAN is a research assistant in the Human Factors Division of the Naval Training Systems Center. He has been with the Human Factors Division for over a year and is currently on an educational fellowship from the Department of Energy. Specific work experience has been in the following areas: evaluation of visual display parameters for computer-based training and aiding applications, computer-assisted instruction, and developing training aids for performance in stressful environments. Mr. Mulligan received his B.S. degree from Florida State University in 1986, and is currently working on his M.S. degree in Industrial/Organizational Psychology at the University of Central Florida.

COMPUTER AIDED TRAINING DEVELOPMENT SYSTEM (CATDS)

J.D. Jared

Boeing Military Airplane Company
Integrated Logistics Support
Seattle, Washington

ABSTRACT

The newly developed Computer-Aided Training Development System (CATDS) is an innovative approach to reducing the time and expense inherent in the Instructional Systems Development (ISD) process. CATDS is unlike other systems in its flexibility of applications, support of user definitions and ability to interface with Logistics Support Analysis (LSA) databases. The overall goal for the system was to provide better training to DoD customers at a lower cost.

CATDS was written in Turbo Pascal to take advantage of its data manipulation speed and practical use on standard PCs. The system currently uses five major files to support task and training requirements analyses. These are: Task File, Definition File, Index File, Equipment File, and Reference File. These are a combination of user-modified and system-modified files and form the main database for CATDS. In addition to the five main files, CATDS supports the concept of task planning matrices to be used during the task identification phase. The analyst inputs and manipulates data through a series of screens.

CATDS generates management and contractual reports through the successive stages of ISD, and from proposal analysis, to final deliverable courseware and training device requirements, including CDRL items. It provides analytical documents and audit trail documentation for any portion of the ISD process. Information available to management enables them to track progress and identify potential problems quickly.

CATDS has been used effectively to support contractual requirements and proposal efforts for aircrew and maintenance training. CATDS has been used to support the A-6 Replacement Wing program, where over 3,000 tasks were analyzed. CATDS has supported the Egyptian and Italian 707 Tanker programs with approximately 1,500 and 1,200 tasks analyzed. CATDS has been proposed for the Advanced Tactical Fighter, the Army's Light Helicopter (LHX) family, the Space Station, the Facility Intrusion Detection System (FIDS) and additional tanker programs for Brazil, Australia, and Spain.

BACKGROUND

The Instructional System Development (ISD) process has been used in its many guises to varying degrees in training system development since the late 1960's. The analysis effort expended in developing training systems has been characteristically a labor-intensive process; the analyst systematically accomplishing most of the tasks manually. The Navy analysts used the procedures in MIL-T-29053B, Requirements for Training System Development, the Army analysts used the Systems Approach to Training Model (Figure 1) and the Air Force analysts used the Instructional System Development Model (Figure 2). Corporations used their own variations of the model, such as the one in Figure 3. These approaches all had one thing in common, they required a lot of "stubby pencil" effort. A significant amount of the training analyst's time was occupied with recording data or compiling reports, all of which were (and still are) predominantly administrative functions.

FIGURE 1. SYSTEMS APPROACH TO TRAINING MODEL

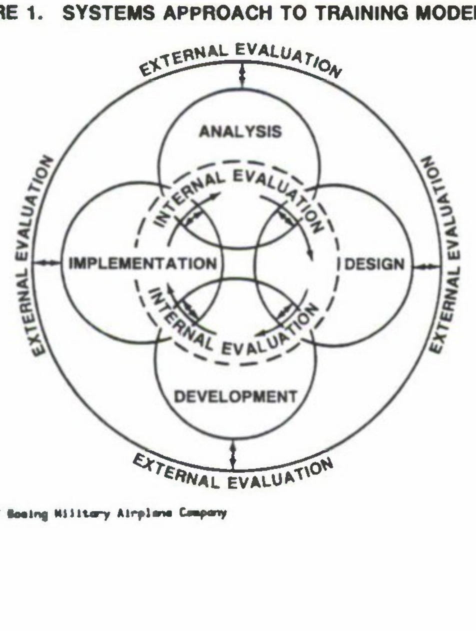
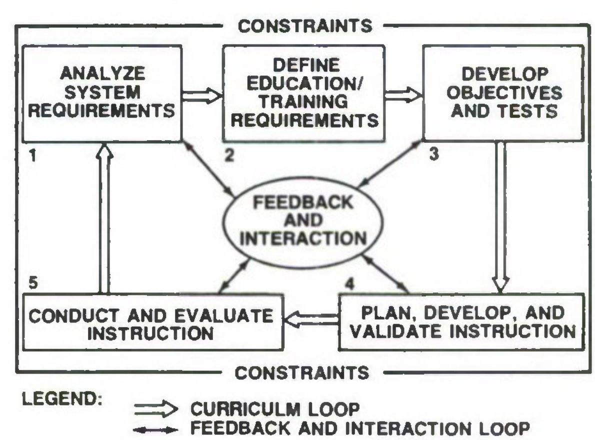


FIGURE 2. INSTRUCTIONAL SYSTEMS DEVELOPMENT MODEL



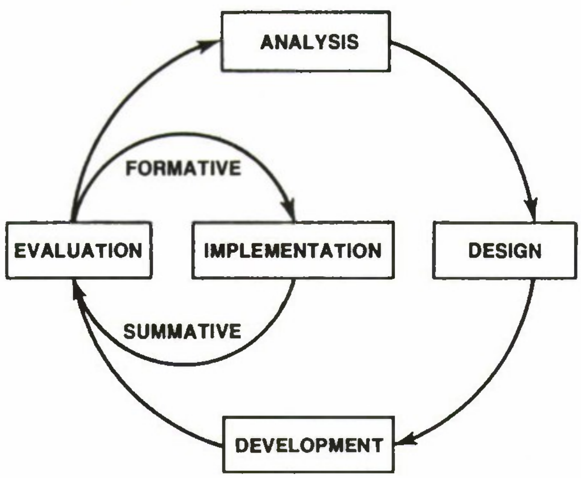
TRAINING ORGANIZATION NEEDS

In order to efficiently meet the needs of a training organization, the analysis system must provide the capability to store and manipulate

information about the following entities:

- Tasks
- Personnel
- Equipment
- References
- Hazards
- Skills and Knowledge
- Cues
- Objectives
- Courses
- Media

FIGURE 3. COMMERCIAL TRAINING SYSTEM DEVELOPMENT MODEL



The training analysis and development systems currently in use have one thing in common. They are all concerned with the identification of tasks that individuals or groups are required to perform and how to implant the requisite skills and knowledge. There is a considerable volume of data required to be generated about each task, even if it "falls out" during the training decision process. Figure 4 identifies the type of information about tasks and steps that may be needed by the analyst, subject matter experts and management during the ISD process.

FIGURE 4. TASK AND STEP INFORMATION REQUIREMENTS

<u>Task</u>
<u>Task Identifier</u>
<u>Taek Description</u>
<u>Task Analysis Information</u>
(<u>From Subject Matter Expert</u>)
Work Unit Code
Work Center
Personnel Requiremente
Taek Frequency
Time Required to Perform Task
List of Equipment Required to Perform Task
List of Refersnces Rsquirsd to Perform Task
Task Cuee
Job-Site Hazards
to people
to equipment
Skills Required to Psrform Task
Knowledge Required to Perform Taek
Proficiency Required to Perform Taek
Task Conditione
Task Standarde
Task Criticality
Task Difficulty
Task Complexity

<u>Training Analyeis Information</u>
(<u>From Training Analyst</u>)
Entry Skills
Entry Knowledge
Entry Proficiency
Training Decieion
Learning Categoriss
List of Appropriate Training Media
Time Between Training and Doing
Training Hazards
to people
to equipment
List of Training References and Documents
List of Training Equipmsnt and Facilities
Objsctivee
List of Instructional Setting (Courses)

<u>Management Information</u>
Taek Status
Hardware Statue
Publications Statue
Date Task Was Creatsd
Date Task Wae Last Changed
Training Analyst Responsible
Generation Number

TREND TOWARD AUTOMATION

Recent developments in computer technology have made available large amounts of data processing capability to a broad variety of users. This increase in capability has resulted in collateral systems which have been automated, such as logistics support analysis, or are rushing towards automation, like technical manuals in the Automated Technical Order System (ATOS) concept. Many analysts have seen the comparable need to apply the power of the computer to training analysis. Some improvements in analytical efficiency have been noted in recent years with the proliferation of personal computers. Primitive computer-based training analysis systems were typically stand-alone systems, not interfaced with source data systems, but merely placed the paper and pencil processes onto the screen and printer. The training analysts used the computer to record their data and to prepare reports. This advancement greatly facilitated the training analysts' ability to manipulate the stored data. However, it remained largely a manual operation, since the data was still loaded keystroke by keystroke. The keyboard merely replaced the stubby pencil. It seemed significant advances in automation were being made in every field except training analysis.

CATDS CONCEPTS

The Boeing Computer Aided Training Development System (CATDS) was conceived as a means to assist the training analyst in the collection, recording, manipulation, analysis, and output of the training data. It was designed to satisfy the requirement for a training analysis and development capability which could support both aircrew and maintenance training in a total system concept. It was determined early-on that the system could not and should not replace the expertise of the training analyst. Any models developed for the system would be recommendations only, with the final decision being made by the training analyst. The model decisions would not be recorded without the concurrence of the analyst.

The development of CATDS had five specific objectives:

1. To computerize the task analysis process while retaining active involvement of the training analyst.
2. To improve the efficiency of the ISD/SAT process by reducing the man-hours required for manual recording and manipulation of data.

3. To provide a standardized, systematic means to approach task analysis.

4. To reduce duplication of effort by providing the analyst immediate access to all task analysis data.

5. To make use of Logistics Support Analysis (LSA) data.

Since the objectives of most training analysis systems are to identify tasks that require training and then determine how to go about training people to do them, CATDS focuses on tasks. CATDS uses existing task lists, tasks from LSA or other sources, or it can be used to develop a task list from an equipment list. A wide variety of task-specific data is collected and stored in a task file by CATDS.

From the practical viewpoint, CATDS had to be relatively easy to use and preferably usable on Personal Computers (PC). The latter requirement provides a significant challenge to store all of the data associated with training analysis and courseware development. Despite our target system (IBM PC/XT) having 640 Kbytes of RAM and 20 Mbytes

on a hard disk, past experience had shown this could be quickly filled, particularly if more than one weapon system was being analyzed at the same time. An alternative to merely storing large ASCII files had to be developed and implemented. In addition, it was felt the programming language should be capable of being transported between different manufacturers' computer systems. Since it had been previously determined that the system would be directed at PC systems due to the generally widespread availability of these machines, we concentrated on a language suitable for use on most of the varieties. Turbo Pascal was selected as it met almost all of the design criteria. It would run effectively on most PC-DOS/MS-DOS compatible machines.

CATDS DEVELOPMENT

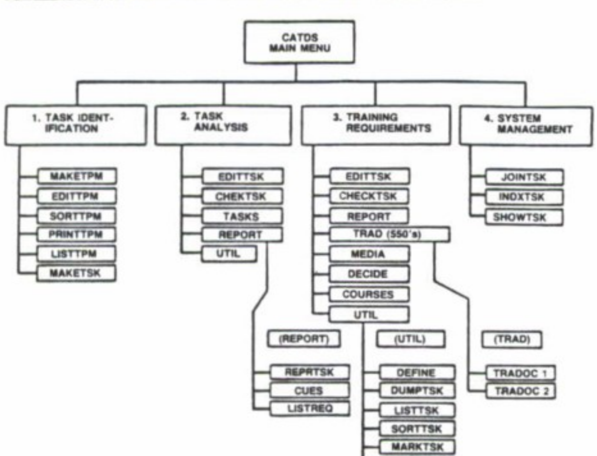
The development of CATDS was a complex project, requiring considerable computer expertise. We requested expert assistance from Boeing Computer Services (BCS). BCS had been working with the BMAC Training and Manpower organization on development of Artificial Intelligence (AI) applications to training. The AI approach maintained data on equipment states by using Boolean matrices, or truth tables. This has proven to be a highly efficient method for storing large amounts of data with small memory requirements. The AI program was a maintenance diagnostic training program using video disc, written in Turbo Pascal and operated on a personal computer. The data storage concept was modified somewhat when applied to CATDS. A schema was developed that uses the first 65 data columns to store relevant information about the task. The stored data might look like this:

150000020500000000000000001234678510-1234TVW CEU

This entry would be followed by the task description. Each element in each position has a discrete meaning, translated by correlation to lines of text in a definition file.

In an effort to make CATDS easy to use [for training analysts and Subject Matter Experts (SMEs) who are not normally computer experts] a menu-driven user interface was developed. It is hiearchical in nature, as shown in Figure 5, with the major elements of Task Identification, Task Analysis, Training Requirements, and System Management forming the intermediate menus leading to the actual programs that make up the heart of CATDS.

FIGURE 5. CATDS USER INTERFACE STRUCTURE

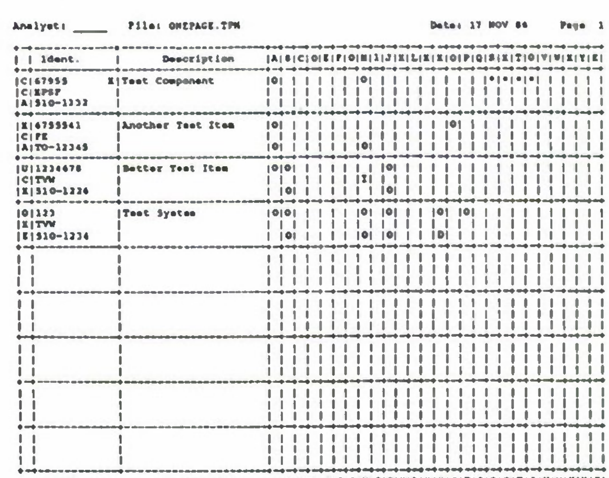


TASK IDENTIFICATION

In the task identification process, CATDS supports several approaches. If a task list is available from another source, such as technical data, subject matter experts, or LSA, it will support completion of the analysis process. CATDS uses a simple seven-step approach to set up task data files. First, a text file of the tasks is created using a text editor or word processor, with the only constraint being a limitation to 79 total characters on a single line. Next, the text file should be reviewed for completeness. The third step creates a new task file through the use of the CATDS MAKETSK program. Provisions are included to identify task steps. The fourth step uses the CATDS program TASKS to print the created task list. The fifth step uses the CATDS program INDXTSK to create an index file. In the sixth step, the program INITTSK is used to set initial values for various data items, if required. The final step is a transition to the analysis phase.

However, in the early stages of a weapon system development, the only available data input is often an equipment list. This requires additional steps to convert the input data into a form usable by CATDS and the analyst. First, a text file of the system or equipment items is created using a text editor or word processor. Next, as before in the use of task lists, the list is reviewed for completeness. The CATDS program MAKETPM uses the equipment list text file to create a task identification matrix (Figure 6). The task identification matrix is used by the program EDITTPM to allow the SMEs to add item-specific information to this file. The fifth step permits the analyst to print out blank copies of the Task Identification Matrices for use by the SMEs in recording their inputs in an off-line mode. The sixth step is the use of the EDITTPM program to enter the SMEs' inputs into the file. The programs LISTTPM and PRINTTPM are used in the seventh step to create review copies of the Task Identification Matrices. The eighth step uses Option "T" of program MAKETSK to generate a task file. The remaining steps are concerned with review, creation of an index file and transitioning into the analysis process.

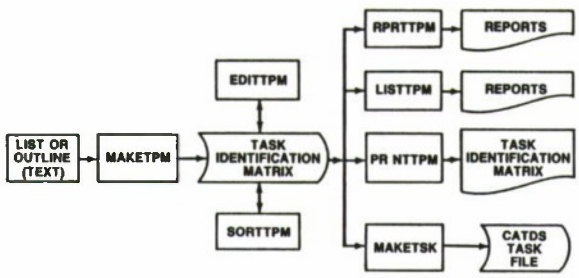
FIGURE 6. TASK IDENTIFICATION MATRIX



The foregoing process for identifying tasks (Figure 7) seems clumsy when described in words, but in reality, it is quite rapid. Two analysts working for two hours were able to identify over 800 tasks for a proposal document. An added

benefit to this approach is that program MAKETSK automatically handles the writing of the task descriptions; spelling is totally uniform. Experience has shown that the simplicity of specifying tasks by this method causes SMEs to tend to err on the side of overcompleteness, which is a highly desirable consequence in the early stages of a project or proposal effort.

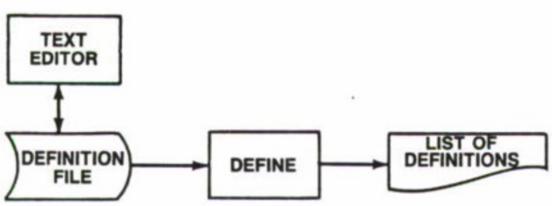
FIGURE 7. CATDS TASK IDENTIFICATION PROCESS



TASK ANALYSIS

During the second step of the training development process, the tasks that have been identified must be analyzed to determine if the tasks require training, and to record task information needed to develop such training. Before this phase of analysis is begun definitions for job codes, conditions, and standards must be placed in a definition file using a text editor, or word processor, as shown in Figure 8. The definition files are used by the system to define text strings to be used in outputs. The four major benefits of using "soft" codes in place of text are: reduced typing labor, reduced computer storage, improved output consistency, and greater system flexibility. Figure 9 shows a sample of the data contained in a definition file.

FIGURE 8. CATDS TASK DEFINITION FILE DEVELOPMENT



Alphanumeric codes are used by the analyst in analyzing the tasks to identify skills (A:-:), knowledge (B:-:), etc. These codes are thereafter associated with the task in the task database and are linked to the definitions in the definition file when outputs are requested. In addition, a reference file and an equipment file should also be established. The reference file is nothing more than a text file containing a list of references. Equipment files are also text files and contain a list of equipment. The analyst uses CATDS program EDITSK to accomplish the analysis.

The use of codes greatly simplifies the data entry requirements for the analyst. All codes starting with an "A" are skill codes, those starting with "B" are knowledge codes, and so on through the alphabet. Each code group has up to 36 possible responses (26 alphabetic and 10 numeric). In some cases, special characters can also be used to represent an entry. Instead of writing/typing text, such as, "Basic FM radio maintenance school", the analyst would enter I:B:. The computer makes the link from the entry to the definition file and automatically converts the code to text in outputs. The analyst saved 29 keystrokes on one entry. The definition file also allows the analyst to reduce the repeated entry of item names. In Figure 9, the use of ### in the verb list is linked to the name of the equipment/item being analyzed. There is a considerable reduction in data entry keystrokes and reduces the requirement for keyboard familiarity on the part of the analyst or SME.

FIGURE 9. DEFINITION FILE — SAMPLE CONTENTS

```
CODE DEFINITION

O:      Navy Sailing Ship Maintenance Training
1:      U.S. Navy
A:A:    Skills gained in "A" school or equivalent OJT
A:B:    Basic skills composite power tools
B:A:    Knowledge gain in "A" school or equivalent OJT
B:B:    Required prerequisite knowledge
C:A:    Given appropriate training device
C:F:    Wings Folded, flaps and slats extended
F:?:    Task frequency is unknown
F:1:    Very infrequently; less than twice a year
G:1:    Organizational level
G:3:    Depot level
I:B:    Basic FM radio maintenance school
J:A:    Avionics Technician (AT)
J:M:    Metalsmith (AMS)
K:1:    No effect on mission
M:L:    Interactive vidodisc
M:S:    Operational trainer
N:3:    Not suitable for aluminum
V:B:    Test ###
V:J:    Repair ###
```

In the use of program EDITTSK, seven different screens are required to complete the task analysis: Basic Task Information; Personnel Factors; Task References; Facilities & Equipment; Conditions & Standards; Task Proficiency Levels; and Job Hazards. The Personnel Analysis screen, as shown in Figure 10, is representative of the task analysis screens. The titles of personnel required to perform the task are entered by the analyst (underlined data), using the "J:" codes in the definition file. The Component codes are used to specify that the task is done by Active Duty, National Guard or Reserve Components. The analyst must use the following codes:

1 = Active Duty
2 = Reserve
3 = Active and Reserve
4 = National Guard
5 = Active and National Guard
6 = Reserve and National Guard
7 = All Components

FIGURE 10. PERSONNEL ANALYSIS SCREEN

```
===========================================================
                    Personnel Analysis

            Title/MOS/AFSC/Rate: A
          Components of Service: 7     [1..7]
            Task Frequency Code: 2     [1..4]
      Percentage of people doing job: 90
Percentage of time spent doing job: 7
             Actual time to perform: 3
                   Task Difficulty: 2

===========================================================
Change Screen: 2
===========================================================
```

The task frequency codes are defined in the definition file. If the data originated from an LSA database, the codes must be interpreted and changed to the standard values as follows:

1: Very infrequently	3: Frequently
2: Infrequently	4: Very frequently

The percentage of people doing the job is derived from surveys, SME experience, or actual user data. CATDS recognizes that not every student will actually perform every task after training. The percent of people who actually do the task is used as part of the basis for deciding whether a task is to be trained or not.

The training analyst uses his knowledge, skills and experience to select codes from the previously developed definition file to identify the environment and requirements of a task or a task step. Similar analyses are used to complete the remainder of the Personnel Analysis screen. The data forms the basis for determining the need for training and who will receive it.

ANALYZING TRAINING REQUIREMENTS

After tasks have been identified and analyzed, the training requirements analysis process identifies those tasks that require training. The CATDS program EDITTSK uses screens "A" through "G" to capture the data and decisions of the training analyst during the training requirements analysis. These screens are:

A. Training Hazards
B. Training References
C. Entry Proficiency Levels
D. Training Decisions
E. Instructional Factors
F. Cues, Facilities, & Equipment
G. Skills & Knowledge

The Training Decisions Screen "D", as shown in Figure 11, is representative of those used during the training analysis process. Training decisions made by the analyst are recorded using this change screen. Referring to Figure 11, probability of poor performance is used as a relative measure of task complexity and is indicated by the analyst according to the following codes:

1: Extremely rare; procedure consists of only one step and no decisions.

2: Rare; procedure consists of one to five steps and decisions.

3: Average; procedure consists of six to ten steps and decisions.

4: High; procedure consists of over ten steps and decisions.

FIGURE 11. CHANGE SCREEN D. : TRAINING DECISIONS

```
=====================================================
                 Training Decisions

Probability of Poor Performance : 1    [1=none .. 4=high]
       Refresher Training Reqmnt: 1    [1 .. 4]
              Task Criticality: 1      [1 .. 4]
           Training Rationales: HEC
             Training Decision: Y      [Y or N]
Months between Training and Task: 1    [0 .. 60]
=====================================================
Change Screen: D
=====================================================
```

Refresher training requirements are a relative measure of the amount of delay that can be tolerated between the time training occurs and the time actual performance would normally take place. Codes and their meanings are as follows:

1: Refresher training can be postponed indefinitely/ OJT sufficient.

2: Refresher training can wait until a subsequent mission/task.

3: Refresher training required for a mission/task.

4: Frequent refresher training required.

A single numeric code indicating why the task is or is not a critical task may be specified. The meaning of this code is defined in the "K" list of definitions in the definition file. Training rationales are reasons supporting or denying the need for formal training. Analysts may specify several codes from the definition file for this purpose. The training decision line in Figure 11 is used to record the final decision as to whether or not training is required. The length of time between training and time of actual performance may be significantly long, which is why there are up to 60 months available in the program.

CATDS has a feature to support the train/no-train decision. Program DECIDE is a rule-based expert system that gives management and analysts outputs for comparison. DECIDE was developed to provide a uniform method of analyzing and presenting train/no-train decisions. DECIDE takes its inputs from several key fields in the task data base, among them training and job hazards, task criticality, task frequency and task complexity.

CATDS will also generate media recommendations. Program MEDIA makes use of expert system technology in selecting media for a training system. The task learning categories previously entered with program EDITTSK are compared to a matrix of media that are the CATDS media list and their respective fulfillment of the specified learning categories. Media are presented for each task in the order of their best fit to the supplied learning categories, as well as the "estimated" media selection entered by the analyst for each task which is presented for comparison. The media considered also include training devices, such as flight and maintenance simulators. The level of detail is suitable for use as functional requirements for training device specification development. In addition, the unique file maintenance aspect of CATDS provides an excellent audit trail back to the initial training requirement identification.

In addition to training and media recommendations, CATDS has the capability to group related tasks into preliminary instructional modules with program COURSE. The resulting modules can be used as course descriptions, as shown in Figure 12, or for further analysis.

REPORTS

For support of system management, CATDS produces reports for program managers and analysts. It has the capability to produce 27 different reports, many of which meet the requirements of MIL-STD-1379B/C, MIL-T-29053B, and other Contract Data Requirements List (CDRL) Data Item Descriptions (DIDs). The partial task listing in Figure 13 is an example of the reports that can be generated by CATDS. Each task and task step has its unique identification number and the structure of the data base that allows the analyst and the system to maintain the internal audit trail required in analysis and report generation.

System management is enhanced by the variety and types of reports available from CATDS. The Manager's Report (Figure 14) identifies the task

and step identification number, who analyzed the task, when it was created, when it was last changed, the generation number, publications status, hardware status, task status, and the model recommendation for train or no-train. This and the other 26 available reports provide management enhanced visibility over the status and progress of the training program development.

FIGURE 12. COURSE DESCRIPTION — GENERATED BY COURSE

```
Course Title: CIN C-602-3761  A-6 Airframe/Hydraulics Drg. Mt. Crse.

Course Objective
  Demonstrate a knowledge, in writing, of
  Demonstrate a knowledge, by performing repairs of

Course Prerequisites
  Skills gained in A School or 6 months equivalent OJT
  Basic skills A-6E existing support equipment
  Basic skills with common aircraft hand tools
  Basic skills with common aircraft power tools
  Demonstrate a knowledge, in writing, of
  Demonstrate a knowledge, by performing repairs of

Personnel
  Corrosion Control (all ratings)                     DOS: None
  (AMH) Aviation STructural Mechanic Hydraulis        DOS: DG-9760
  (PC) Plane Captain (all ratings)                    DOS: None
  Familiarization                                     DOS: None
  (AMS) Aviation Structural Mechanic Structures       DOS: DG-9760
  Test Team

Media
  Reference Books, Manuals or Text (Print)
  Reference Charts
  Overhead Transparencies
  FAM Trainer/Full Scale Mock-up

Course Content

A0S005 Inspect Spar, rear, 18, L&R
V0S005 Preserve Spar, rear, 18, L&R
A0S007 Inspect Spar, leading edge, 18, L&R
V0S007 Preserve Spar, leading edge, 18, L&R
A0S008 Inspect Can, Slat Track, 18, L&R
H0S008 R & R Can, Slat Track, 18, L&R
J0S008 Repair Can, Slat Track, 18, L&R
V0S008 Preserve Can, Slat Track, 18, L&R
A0S009 Inspect Ribs, tank end, intermediate, 18, L&R
...
```

FIGURE 13. LIST OF TASKS AND STEPS — GENERATED BY STEPS

```
15 NOV 86     Task Steps from File: TEST.TSK        Page 1

Inspect Actuator [ADA001]
   1. Remove access cover [ADA001/010]
   2. Visually inspect Test Item for corrosion and cracks [ADA001/020]
   3. Replace access cover tightly [ADA001/030]
   4. Record inspection in aircraft maintenance log [ADA001/040]

Test Actuator Base Plate at depot (NARF) [BDC003]

Test Actuator [BDA001]

Test Upper Hydraulic Line [BDC003]

R & R Lower Return Hydraulic Line [HDC002]

R & R Upper Hydraulic Line [H1C003]
```

FIGURE 14. MANAGER'S REPORT — GENERATED BY SHOWTSK

15 NOV 86 Tasks from task file: TEST.TSK Page 1

Task/Step ID	TA/ SME	Date Task Created	Date Last Changed	Gen. Number	— Statuses — Pub.	HW	Task	Model T/NT
1G1	1st	15 NOV 86	15 NOV 86	0	G	G	G	?
1E1	1st	15 NOV 86	15 NOV 86	0	G	G	G	?
1E2	1st	15 NOV 86	15 NOV 86	0	G	G	G	?
ADA001	JDO	15 NOV 86	15 NOV 86	2	G	G	G	Y
ADA001/010	JDO	15 NOV 86	12 DEC 86	2	G	G	G	Y
ADA001/020	JDO	15 NOV 86	15 NOV 86	2	G	G	G	Y
ADA001/030	JDO	15 NOV 86	10 DEC 86	2	G	G	G	Y
ADA001/040	GJM	16 NOV 86	16 NOV 86	3	G	G	G	Y
BDA001	MEG	15 NOV 86	15 NOV 86	3	G	G	G	Y
RDA001	JDO	15 NOV 86	15 NOV 86	2	G	G	G	Y
SIA001	JDO	15 NOV 86	15 NOV 86	1	G	G	G	N

CAPABILITIES

CATDS has been used by the Boeing Military Airplane Company in support of several programs and for proposal development. Training analysts have used CATDS in the analysis of over 3,000 tasks in support of the A-6 Replacement Wing program for the U.S. Navy. In the Egyptian and Italian 707 tanker/transport aircrew and maintenance training programs, approximately 1,500 and 1,200 tasks were analyzed respectively. The manhours required to accomplish the training analysis and program development were reduced by approximately 30 percent over previously used systems. This savings includes the lower portion of the learning curve for the analysts.

CATDS was used during the A-6 Replacement Wing program to identify maintenance training equipment fidelity requirements. These requirements, in turn, were used in the development of Prime Item Development Spedifications (PIDS).

A significant benefit to the training analyst has been the use of CATDS in proposal development. There is a quantum step forward in the amount of detail that can be incorporated into a proposal. In a proposal to develop training for the Australian 707 Tanker/Transport RFP, two analysts working only two days identified and analyzed 1,008 tasks. CATDS was used to analyze these tasks and create preliminary course outlines. These course outlines were included in a Tentative Training and Training Equipment Plan (TTEP) which was included with the proposal.

CATDS has used task data obtained from the LSA database for training and training requirements analyses. In preliminary preproposal work on the Army's light helicopter (LHX) program CATDS was used to extract tasks directly from the LSA data base, specifically the C06 Records. This was accomplished on two separate occasions to verify the effectiveness of the process. It has also been used to obtain and analyze tasks from the A-6 LSA data base. With the experience of accessing LSA data, it is not too difficult to envision access to paperless publication databases.

SUMMARY

CATDS is an effective and efficient tool for the training analyst to develop training programs for operator and maintenance personnel. The unique data file management capability of CATDS provides a clear audit trail for courseware and training devices. The report capability alerts training analyst to changes in hardware design or training requirements. Analyst's decisions can be compared to computer-generated models. The use of CATDS reduces man-hour expenditures in proposal development, training analysis and training development, contributing to reduced life cycle costs. The various reports provide for greater management visibility of program status and progress. CATDS operates on standard PCs. It retains analyst's unique expertise by requiring all decisions to require approval prior to finalization. It enforces adherence to analytical procedures by standardizing the methodology used by the analyst on each task and program. CATDS has demonstrated compatibility with LSA standards.

ACKNOWLEDGEMENTS

Mr. Gary McNeil, Boeing Military Airplane Company, Wichita, KS, was one of the main initiators of the CATDS development effort. His understanding of training systems aided immeasurably in the system effort and in this paper. Mr. John Howell, Boeing Computer Services, Wichita, KS provided the artificial intelligence expertise and technical capability to blend the various training analysis and development requirements into a cohesive and usable system.

ABOUT THE AUTHOR

Mr. J.D. Jared is a Training Analyst with BMAC's Integrated Logistics Support organization in Seattle, WA. He is the training and technical order lead for the BMAC Advanced Tactical Fighter (ATF) program effort. He was recently reassigned from BMAC's Simulation and Training System organization in Huntsville, AL. He holds an MBA degree from Harvard University and a BA degree in Radio-Television from the University of Washington. He was recently enrolled in a Computer Information Systems degree program at Athens State College, Athens, AL.

CONSTRUCTING AN INTELLIGENT TUTOR USING AN EXISTING EXPERT SYSTEM (PILOT'S ASSOCIATE)

Ruth M. Hawkins
Robert W. Ensey
Sue Kemner-Richardson
Douglas Aircraft Company
Training Systems and Services
Computer-Based Training Systems
Aurora, Colorado

ABSTRACT

Military research organizations (e.g., HRL, DARPA, ARI) have been funding efforts to design and build expert systems to aid in the maintenance and operation of weapon systems. As the technology matures and these expert systems become more practical, it may in some instances be possible to use an existing expert system knowledge base as the expert module in an intelligent tutor/coach. The authors begin by providing a brief introduction to Expert Systems (ES) and Intelligent Tutoring Systems (ITS), including a discussion of the advantages derived from using an existing ES as the basis for an ITS expert module. They go on to discuss the degree of cognitive fidelity that exists in the expert system as a factor to consider when designing the ITS. Finally, they describe a specific example where this approach may be feasible--the Pilot's Associate.

INTRODUCTION

Objective

In this paper, we will explore issues that arise when using an expert system (ES) as a base for developing an intelligent tutor or coach. Both expert systems and intelligent computer-assisted instruction are products of the past decade of research in artificial intelligence. Expert systems seek to replicate human performance requiring expertise in a narrowly defined domain. Intelligent tutoring systems (ITSs) seek to teach as human tutors or coaches do. A tutor needs to know what is being taught; he or she must be a domain expert. Therefore, the linkage between expert systems and ITSs is that perhaps the expertise captured in the expert system could serve as the expertise required of a knowledgeable tutor. Can this approach work? And why or why not? This paper attempts to answer these questions.

Increased Performance Requirements and ES Technology

The next two decades bring special challenges to training research and development. In military training, increasing weapon system complexity coupled with a decreasing pool of recruits means that there will be fewer personnel to perform increasingly complex tasks.

One of a number of responses to this potential crisis, in the maintenance domain, has been the introduction of increasingly complex automatic test equipment. Another sophisticated and more recent response has been the advent of expert systems technology in maintenance and operations. Expert systems have been used to perform the actual work itself, e.g., diagnose, decide, plan, etc. They have also been used as a kind of high-tech job aid for personnel in the performance of their jobs. Related to these ES endeavors is the potential increase or decrease in need for training. The dichotomy in technical training has been "smart operator/maintainer-dumb machine" versus "dumb operator/maintainer-smart machine".[5] With the potential use of new, intelligent training technologies, the dichotomy disappears. The machine and the human can work together to achieve success. In effect, with the ES assuming more of the pedestrian activities, the human is freer to learn more about the domain and become more expert than the machine at higher-level or more complex tasks.

Description of an Expert System

An expert system is a computer program that has two fundamental components (as most computer programs do): the program and the data. We will describe the latter of these two components first.

The data include two structures: the knowledge base and working memory. The knowledge base consists of a collection of IF-THEN rules, and the working memory contains facts about the outside world. These facts are provided in response to a question of the user, an interface to another information system, or are deduced by the ES itself based on other information it has. For example, suppose an expert system exists that provides advise on choosing a national park for your vacation. A rule in the knowledge base might be "IF caves are most desired feature, THEN select the Carlsbad Caverns". Within working memory, the ES contains information about the interest the user has in caves based upon the user's answer to a question.

The program, often refered to as the "inference engine" or control strategy, chains or strings the rules together so that the presence of data in working memory can be used to generate new data through the inference represented by the rules.

The core work in building an ES is deciding both what the basic facts of a domain are and how these facts are related through rules. This activity is called knowledge engineering and results in a knowledge base. The inference engine is largely domain independent. Providing an ES with different expertise is largely a matter of changing rule bases.

ITS Technology

In contrast to traditional computer-assisted instruction (CAI) characteristics, an ITS can generate and answer questions, support a rich variety of user responses in accordance with an instructional strategy, and model the student not in terms of observable behavior but in terms of the underlying reasoning which must be present in order to produce that behavior.

An ITS can be viewed as being comprised of three modules:

1. Expert module
2. Student module
3. Tutor module

These three modules exist within an instructional environment and with the use of a communication interface.[6]

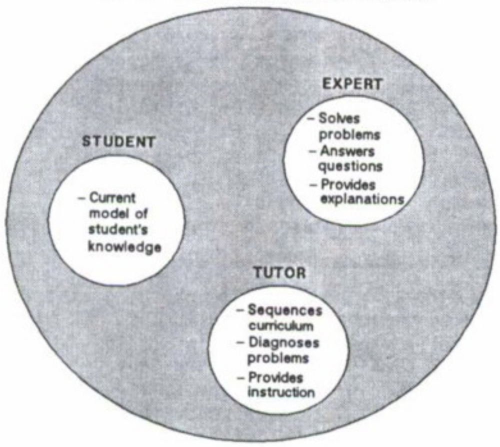


Figure 1. ITS Modules, Environment and Interface

The expert module can solve problems, answer questions, and provide explanations in a domain. The student modeling (or diagnostic) module can infer what aspects of an expert's knowledge the student has. The tutor module can sequence the overall flow of topics (the curriculum), provide instruction, including the selection of specific problems to present, and provide help, hints, advise, or explanation in response to queries from the student. All of these modules must be coordinated in the context of some instructional environment which serves as the medium of communication between the student and the ITS. Environments vary ranging from unstructured discovery to a very highly controlled and guided environment.

Advantages of Using Existing Expert Systems Within an ITS

The knowledge engineering process to develop domain knowledge has been estimated to be approximately 50 percent of the total effort required to develop an intelligent tutoring system.[2] When a knowledge base for a given domain has been developed, an opportunity exists to integrate its knowledge base in the expert module of an ITS. If this proves feasible, some portion of the cost of developing knowledge bases can be avoided when building an ITS for which an ES exists.

A second potential advantage of reusing an existing ES is the possibility to integrate training with job aiding. Job aiding, provided through the expertise of an expert system, is generally aimed at prompting the performance of the person using it.[5] If expert systems can be enhanced with student and instructor modules, they can do more than merely prompt performance: they can serve as an on-the-job training vehicle. Such integrated aiding and training is extremely attractive. However, it is recognized that expert systems, as the product of human effort, are apt to embody a level of excellence below that of genuine human experts. Here two things follow. First, training remains an important requirement in order to maintain the existence of human experts, and second, human experts are required in order to maintain and enhance the knowledge base of the machine expert.

THE MACHINE EFFICIENT =| COGNITIVE MODEL CONTINUUM

The goal of an intelligent tutoring system is to convey expertise to a human student. The experience of a number of ITS researchers[2,7] shows that the more closely the expert module reflects human cognition, the easier it is to achieve that goal. Therefore, when evaluating an existing expert system for use within an ITS, two questions arise-- How explicit is the expression of the knowledge and is the reasoning process human-like?

As expert systems move on a continuum from a "black box"[2], machine efficient model toward a cognitive model, the possibilities for depth in tutoring increase. For the purpose of our discussion, a black box model refers to an expert system where the input and output are known but the operations performed to achieve the output are hidden. At the other end of the continuum is the expert system based on a deep-structure cognitive model, where the knowledge base is fully developed and is structured in a way that reflects human reasoning. In the following section, we describe techniques that have been used to deal with expert modules that demonstrate varying degrees of cognitive fidelity.

Toward A More Cognitive Model

In order to train students in any discipline, the activities, theories, processes, procedures, and concepts of that discipline must be well understood by the tutor. We are not going to debate whether or not a computer can "understand" these things or to what degree. What is important is how we as system designers can model, in the computer, these distinctive human capabilities. It is reasonable to say that in order to train a person in human activities, this model must be designed to simulate human mental processes. In other words, a cognitive model of human problem-solving must be incorporated into the machine tutor. In many cases the "best" method for a machine to solve a problem is to try all the possibilities, i.e. the exhaustive search strategy used in chess programs. Human problem-solving in most cases involves some other heuristic strategy. By using a cognitive model for tutoring, we should not only be able to train students in a given problem-solving methodology, but we should also be able to help

the student to extend his or her thinking process.

In addition, incorporating an explicit cognitive model in the expert module of an ITS helps toward the goal of an articulate tutor, able to explain each problem-solving decision in terms that correspond to those of a human problem-solver. One problem with which we are faced is identifying inference strategies and knowledge representations that model human cognition and can lend themselves better to an ITS. We shall identify those structures and formalisms that will not only be more amenable to tutoring systems, but will also improve the robustness, performance, and flexibility of the expert system itself.

Figure 2 illustrates the continuum from a purely machine efficient ES implementation to a deep-structure cognitive model. Several points on this continuum are described in terms of the degree to which the knowledge representation and control strategy reflect human cognition. Possible methodologies for incorporating each type of ES into an intelligent tutor are identified and will be discussed in more detail in the following section. At this point, it is important to note that existing ES's use a variety of knowledge representation schemes and control strategies; it truly is a continuum without discrete points.

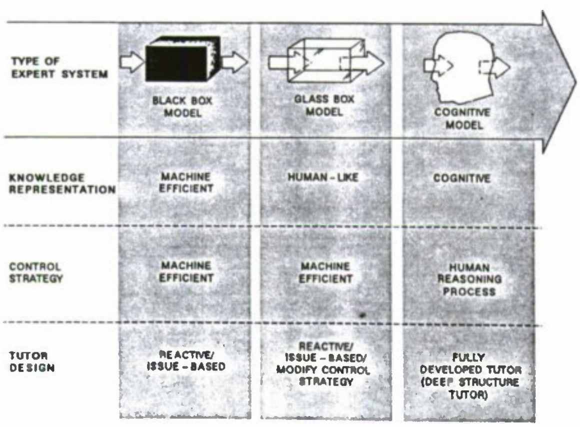


Figure 2 The Machine Efficient →Cognitive Model Continuum

Methodologies

Reactive Tutors Attached To Black Box Model. The most modest form of tutoring system that can be attached to almost any expert system is called a reactive tutor[2]. By inverting the user interface of the expert system (i.e. the student is asked to solve the problem instead of the ES), the tutor can monitor the student's behavior over a range of tasks in the applied domain. The student's performance will be compared with how the expert model would respond and will provide a simple reaction by telling the student whether the response is right or wrong. Depending on the instructional strategy, the reactive tutor may proceed to tell the student what the correct answer should be according to the expert model. The original SOPHIE (SOPHisticated Instructional Environment) effort provides a good example of a reactive tutor; it is a general-purpose troubleshooting simulator which works by solving a set of equations rather than by human-like causal reasoning. This system provides the student with a learning environment where problem-solving skills and ideas may be investigated by the student's own initiative, rather than dictated by instructional sequence.[4]

However, the reactive tutor does not go any further than the right/wrong feedback. It makes no attempt to explain why a response is correct or incorrect; in other words, it is not articulate. It also cannot generate additional questions, or a sequence of queries in order to build up to a point based on the expert model's knowledge. It simply reacts to the situation at hand. In fact, it is questionable whether this type of tutor is an ITS at all. The only intelligence it has comes from the expert system.

Attaching a reactive tutor to an existing expert system may be the most economically and developmentally expedient approach. It is considered by some to be worse than conventional CAI. However, because of its simplicity, broad applicability and low cost, a reactive tutor may be attached successfully to virtually any off-the-shelf expert system, and specifically to the black box expert. In fact, the reactive tutor is one of only two choices for an ITS using a black box expert. This is because the internal computations, search strategy and knowledge representation by which a black box expert provides its contribution to the pedagogical process is either unavailable or inappropriate for human reasoning. The reactive tutor does not require access to the internal components of the expert system. Another important characteristic of the reactive tutor is that it lays the groundwork for the development of an issue-oriented ITS, the next level on our continuum.

Issue-Based Recognizers Attached To Black Box Model. Perhaps a more pedagogically sound tutoring methodology for an ITS is an issue-based tutor. The goal here is to build a more articulate tutor around an expert system that does not allow meaningful access to the knowledge base of that system. In order to do this, the tutoring system incorporates two elements within the expert system.

First, a differential model compares the student problem-solving behavior and performance with the behavior and performance of the expert system given the same situation. The student's model represents his or her knowledge at a particular time, and is considered a subset of the expert's knowledge. A differential is constructed between the two which may suggest hypotheses about what the student does not know or has not yet mastered. A reactive tutor may be used for this component. In order to construct this differential model, the system must:

1. Evaluate the student's current behavior and performance with respect to a set of possible alternative actions that are possible under a given set of circumstances. These actions are determined by the expert system and are considered optimal.

2. Determine what underlying skills the student must have used to arrive at his or her decisions which led to a particular action. Each alternative action by the expert must be analyzed in this manner. For further information on differential modeling, see the article on WEST in the Handbook of AI.[4]

Second, an issue recognizer identifies issues or concepts that are currently relevant in the problem-solving process. This component is data driven and identifies patterns in the differential model and attaches instruction to those patterns. Depending on the complexity and instructional design of the domain, this instruction can be prefabricated ahead of time, or constructed ad hoc by the expert module. Examples may also be constructed using a simulation script to be run through the expert system to show the student an alternative action. These examples provide concrete instances of the abstract concepts (the issue) being taught.

Although an issue-based tutoring system is less tractable and more difficult to construct than a reactive tutor, it is more articulate in relation to instructional objectives. The power of the expert system can be exploited in greater detail and represents a better use of resources. A problem with issue-based tutoring systems, however, is that they can be used for only surface-level (shallow knowledge) tutoring. Any deep-knowledge tutoring would require access to the internal reasoning and knowledge structure of the expert system. A cognitive model for problem-solving does not distinguish this hierarchy so rigidly. The human mind brings all these resources--short and long term memory, shallow and deep knowledge--and applies them to the solution of a problem. We would consider the issue-based tutoring system to be the method of choice for attaching an ITS to a black box expert system model. Additionally, it should be noted that the issue based tutoring system is not limited to the black box model, and could be used with virtually any type of expert system.

The classic example of the issue based system is WEST, developed by Burton and Brown, based on the children's math game "How the West Was Won." In WEST, the student is involved in playing the game, while the instructional program "looks over his or her shoulder," and occasionally offers criticisms or suggestions for improvement. The intention is to coach the student without interrupting so often as to become a nuisance and destroy the student's fun at the game. WEST attempts to guide the student's learning through discovery.[4]

Glass Box Expert Systems With A Human-Like Knowledge Representation. The next level of tutoring methodologies on our hierarchy attaches an intelligent interface to an expert system that uses a more cognitive knowledge representation. Glass box model expert systems have knowledge that is accessible but does not reflect the depth of knowledge required to explain the domain to a novice/student. These expert systems are typically built using a knowledge engineering process to construct the knowledge base. A human domain expert works with a knowledge engineer to formalize the processes, concepts, facts, heuristics, etc. for the machine expert. The nature of this process produces an expert system that has a more articulate, cognitive knowledge representation than that found in a black box model. In addition to having the internal knowledge representation of the expert module accessible to the tutor (glass box), the knowledge representation should model the domain knowledge in a manner that parallels human thought. One problem that the knowledge engineer may face in trying to create a cognitive knowledge representation is that this representation must be appropriate to the domain.

Semantic nets are often used in the expert module because of the implicit inheritance property of the structure (SCHOLAR uses semantic nets to represent its knowledge base.[4]) A semantic net represents abstract objects as a node and a directed graph, and relationships between objects as links between nodes. It is ironic that the very characteristic that makes semantic nets so attractive, inheritance, is its downfall when it comes to modeling human-like knowledge. For example, the following figure states that Dave is a programmer, and a programmer is a "hi-tech" profession, and a hi-tech profession is paid a high salary. This rigid structure cannot take into account the fact that although Dave should, he may not be paid a high salary.

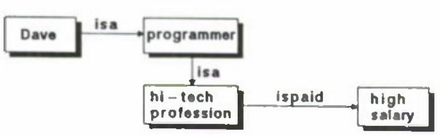


Figure 3 Semantic Net Example

Obviously, the simple semantic net does not model the complexity of human thought. It falls short when we use inference with the knowledge structure. Furthermore, there is no distinction in the network formalism between an individual and a class of individuals.[3] This simple inheritance scheme is too dogmatic, and is not an optimal representation for a cognitive approach.

A more suitable knowledge representation is an extension to the semantic net, called frames. Frames may be considered a super set of semantic nets; they have inheritance properties, but are much more flexible and robust. It is generally accepted that people use a large, well-coordinated body of knowledge to form memories of their previous experiences. These are then used to interpret new situations. For example, when a friend says "I went to a concert ...," we instantly create a mental

representation of an entire set of expectations dealing with the domain of concert going: a crowd of people, a large auditorium, an evening event, etc. We may, in fact, construct a generic template, with a slot for style of music, dress, lighting, cost, etc. Once we determine a more specific context for the concert, our minds fill in these slots with information from past experiences.[3] This is similar to how frames work.

A frame is a knowledge structure that is used to describe values of attributes for an object. These attribute values are stored in slots. Frames may be linked together in a hierarchial manner in much the same way as semantic nets. The power of frames comes from these slots. A slot can contain a value for the attribute, a procedure for calculating the value (an algorithm), or production rules for finding the value (heuristic). A slot may also "point" to other frames that describe the attribute. The structure of the frame is not fixed, so a knowledge structure can be modified as the system "learns." An additional advantage of frames is that the knowledge representation is separate from the inference strategy. This allows the expert module to choose a strategy that is appropriate for a particular domain.

Because frames allow for a more cognitive knowledge representation, they are a good choice for use in our glassbox expert module for an ITS. The flexibility of frames allows the tutor to attach a tutorial agenda to a slot in the frame. The ITS can then build a student model based on how the student has filled in each slot. Then, by filling in the missing information, the ITS can begin to debug or correct student misconceptions.

In general, expert systems based on a glass box model would be candidates for the same tutor design strategies as a black box model--reactive and issue-based. In addition, there is the potential for enhancing the cognitive fidelity of the expert system by modifying the knowledge representation to achieve a deeper knowledge structure or the control strategy to more closely emulate the human reasoning process. This would only be possible if the knowledge representation is truly independent of the control strategy (e.g. a frame-oriented knowledge representation).

Expert Systems with Deep-Structure Knowledge Representation And Human-Like Control Strategy. The problem of constructing a cognitive expert system for an ITS is not entirely solved by a knowledge representation that parallels human thought processes. The knowledge base may reflect the human expert's knowledge but is inadequate to provide the depth of explanation required when teaching novices.

We have all known experts, in a particular domain, who perform well in their area but are unable to convey this expertise to students. When experts solve problems, they often use what is termed "shallow reasoning" or "compiled reasoning", a strategy that is suited for performance or for solving problems quickly.[15] They skip steps in the reasoning process and develop their own peculiar heuristic strategies in order to be more efficient. This form of knowledge is typically built painstakingly over time through experience and the exact rationale for its structure may be difficult to recall.

In contrast, "deep reasoning" describes a sequence of mental states an expert would follow when confronted with a novel, previously unencountered problem, or when providing an explanation to a novice. In these situations, the expert must fall back on reasoning from first principles, i.e., a mental model of the structure of his or her domain, the varieties of declarative knowledge associated with the domain, and a set of problem solving strategies different in character from those of compiled reasoning. Therefore, the highest degree of cognitive fidelity suited for instructional purposes requires the deep form of knowledge that is useful in novel situations and is necessary to generate the most complete explanations for students.

Many expert systems incorporate a non-cognitive control strategy, particularly those that are developed with an expert system building tool that uses a built-in control strategy. Prolog, a programming language for building expert systems, uses backward chaining and cannot be modified. Mycin, a medical diagnosis expert system, also uses backward chaining control strategy. As we shall discuss below, this was a major obstacle when attempting to attach an intelligent tutor to Mycin. Backward chaining may be efficient for a computer, but when trying to coach a human problem-solver in the same domain, it will not help the student to gain added expertise in his or her own problem-solving abilities.

Although it may be quite understandable, humans do not, as a general rule, use this type of control process. Backward chaining involves reasoning backward from goal states to the initial state (working backward from a theorem to axioms for example). This was made evident during the early years of AI by Newell and Simon in the development of GPS (the General Problem Solver). In the formulation of this system, a new control strategy was developed call means-ends analysis. This strategy uses both forward and backward chaining. Not only is this more cognitive, but it turned out to be much more efficient in terms of computer time.[8] Other cognitive strategies are opportunistic, hill climbing, and heuristic search algorithms such as the A* best first search.[14]

The importance of depth in the knowledge representation and a human-like control strategy was demonstrated in the development of Neomycin from GUIDON and Mycin by W. J. Clancy.[4] Mycin is a well known medical diagnosis expert system which was developed using a formal knowledge engineering process to construct the knowledge base, and using exhaustive backward chaining as a control strategy. There were two main problems encountered when Clancy attempted to create a diagnostic tutor (Neomycin) using Mycin:

1. the knowledge base consisted of highly compiled rules that performed well for diagnosis but were not explicit enough for teaching purposes.

2. the control strategy (backward chaining) was incompatible with the human reasoning process.

Because of the shallow articulation of diagnostic rationale and non-human-like control strategy, explanations were very difficult for the machine tutor to generate. As a result of these difficulties, Neomycin was designed with a more explicit knowledge base and a different control structure (forward inference) was imposed. Clancy discovered that an effective tutoring system must be concerned with how the domain knowledge is deployed as well as the content of the domain knowledge.[2] Also, the rules that were used to construct the production system of MYCIN and GUIDON were too highly compiled, i.e. the way in which the knowledge was represented was too shallow. The system essentially had to be pulled apart in order to make to control strategy separate from the rules.

PILOT'S ASSOCIATE (PA)- A POTENTIAL APPLICATION AREA FOR ITS

General Description of PA

Our discussion to this point has been generic in nature, referring to the advantages of incorporating any existing expert system into an intelligent tutor or coach. However, our review of the literature in this area was specifically an attempt to identify a methodology for evaluating a particular set of expert system software for ITS development. The knowledge based system of interest is the Pilot's Associate. The Pilot's Associate Program is sponsored by the Defense Advanced Research Projects Agency (DARPA) and is managed by the USAF. Two industry teams are under contract to develop the Pilot's Associate (1) McDonnell Aircraft and Texas Instruments and (2) Lockheed-Georgia supported by a number of other companies.

As the name suggests, the Pilot's Associate (PA) will serve as the "phantom crew" for post-1995 fighter pilots, providing expertise in mission tactics, air combat tactics, aircraft systems, weapons, targets, and threats. Using a sophisticated pilot-vehicle interface, it will manage and provide status/advisory information to the pilot, interpret the pilot's intent, and assess the pilot's physical/sensory capability.[13]

Implications for Pilot Training

The introduction of PA into the cockpit has tremendous implications for pilot training. Even though PA can assume much of the cockpit workload, the pilot will still need to acquire knowledge that is consistent with the PA knowledge base. From an acceptance perspective, pilots may be distrustful of the data they obtain and the ability of PA to perform. Using portions of the PA knowledge base within the context of an intelligent tutor or coach would provide a number of benefits in a combat aircrew training situation:

- The content taught would be standardized and consistent with the PA knowledge base the student will ultimately rely on during an actual mission.

- The amount of classroom and simulator instructor time could be reduced, which would lower training system life-cycle cost.

- An intelligent tutor or simulator coach can make PA knowledge and reasoning more explicit to the student pilot during ground training, preparing him to accept PA input more readily later, inflight, where explanations are necessarily concise.

- Although ITS technology is new, it has been demonstrated and appears to provide training that is more like that delivered by a human instructor. Therefore, there is the potential for higher quality training to occur. Theoretically, the type of training that can be delivered using intelligent CAI will result in deeper processing of information on the part of the student than can be achieved with conventional CAI or even with some less experienced or skilled human instructors.

What form would an intelligent PA tutor/coach take? For example, it might be used to conduct a mixed initiative dialogue with the student pilot to guide him through the process of "debugging" his misconceptions about aircraft systems. A tutorial module like this might be delivered on an inexpensive computer terminal. A more complex implementation might be an intelligent simulator tutor/coach that observes the student pilot's behavior and generates scenarios based on the current student model. It would make inferences about faulty intent or understanding and provide qualitative feedback, advise, and explanation. In fact, the implementation of this intelligent aircrew training software, based on the evolving PA expert systems, could have an impact on pilot training before the post-1995 introduction of PA to the cockpit. Current ground training--classroom and simulator--could experience the benefits described above if the feasibility of this approach is demonstrated.

Prototyping a PA Tutor/Coach

As mentioned before, ITS technology is new. Until recently, those involved in ITS development have been primarily concerned with answering basic AI research questions rather than practical application of the technology. However, we are now beginning to see the development of ITS's for use in applied settings, for example, Anderson's Lisp Tutor[1], Wolf's Recovery Boiler Tutor[16], and the Maintainer's Associate. In fact, we are aware of at least one on-going effort to embed one of these existing ES's (Maintainer's Associate) into an intelligent instructional environment.[10] Although the technology is still immature, we believe that a prototype ITS based on the PA concept is feasible.

To begin this effort, the first step would be to analyze, from a top-level perspective, the

impact PA would have on pilot training requirements given the current, preliminary design of the system. This would involve review of PA requirements and design documents and discussions with PA engineers and military fighter pilot training experts who are familiar with the program.

Once we better understand the impact that implementing PA will have on pilot training, we can evaluate the functions and subfunctions within PA for suitability as an ITS content area. The criteria for determining suitability can be categorized into at least two different areas.

First, a suitable PA function would need to be an area that is likely to be a training requirement. For example, the PA may include an expert system to perform in-flight diagnostics but it is unlikely that the pilot will need to acquire that particular set of skills and knowledge. He may, however, be required to understand the reasoning that occurs within PA in the emergency procedures area so that during inflight emergency situations, he can quickly evaluate and place confidence in the advice he receives from PA.

The second major area is concerned with the extent to which the knowledge base and control strategies can be incorporated once a particular functional area is identified as training relevant. Each candidate portion of expertise within PA would need to be examined in light of the machine efficient =| cognitive model continuum described earlier.

CONCLUSION

Obviously the design, development, and integration of a truly intelligent tutoring system is a tremendous task. In this paper we have outlined some of the benefits of using an existing expert system to save a certain amount of this effort. However, "off the shelf" expert systems may not be incorporated without some additional effort. In fact, one may need to modify the control strategy or use a totally different search method for the tutor than what the expert module uses. Although we emphasize the importance of the existing expert system's cognitive fidelity, there are other issues to be considered. For example, there is the entire set of questions one must ask when considering the appropriateness of training a particular student population in the domain the candidate expert system addresses.

As expert system technology improves, there will be greater potential for using existing systems as the expert module in an ITS. With feasible methodologies for incorporating these existing systems, we should be able to reduce the development time for intelligent tutors. ITS development technology is new and labor intensive but holds much promise for vastly improving the quality of instruction we can deliver with computers.

REFERENCES

1. Anderson, J.R. "Analysis of student performance with the LISP Tutor." In N. Frederiksen (Ed.)(in press). Diagnostic Monitoring of Skill and Knowledge Acquisition. Hillsdale, N.J., Erlbaum Associates.

2. Anderson, J.R. "The Expert Module." In Polson, M. and Richardson, J. (Eds.)(in Press). Foundations of Intelligent Tutoring Systems. Hillsdale, NJ: Lawrence Erlbaum.

3. Barr, Avron and Feigenbaum, E.A. (Eds). "III. Knowledge Representation," The Handbook of Artificial Intelligence, volume I. HeurisTech Press, Stanford, California, 1982.

4. Barr, Avron and Feigenbaum, E.A. (Eds). "IX. Applications-oriented A.I. Research: Education," The Handbook of Artificial Intelligence, volume II. HeurisTech Press, Stanford, California, 1982.

5. Binkin, M. Military Technology and Defense Manpower. Washington, D.C. The Brookings Institution, 1986.

6. Burton, R.R. "The Environment module of Intelligent tutoring Systems." In Polson, M. and Richardson, J. (Eds.)(in Press). Foundations of Intelligent Tutoring Systems. Hillsdale, NJ: Lawrence Erlbaum.

7. Clancy, William J. "Methodology for Building Intelligent Tutoring Systems." In Kearsley, G.P. (Ed), Artificial Intelligence And Instruction: Applications and Methods. Addison-Wesley Publishing Co., 1987.

8. Cohen, Paul and Feigenbaum, Edward (Eds). "XI. Models of Cognition," The Handbook of Artificial Intelligence, volume III. HeurisTech Press, Stanford, California, 1982.

9. Harmon, P. "Intelligent Job Aids: How A.I. will change training in the Next Five Years." In Kearsley, G.P. (Ed), Artificial Intelligence And Instruction: Applications and Methods. Addison-Wesley Publishing Co., 1987.

10. Jones, Kent. Bamford, Deborah. Richardson, Jeffrey. Sizer, Tania. (in Press) Maintainer's Associate Training Instructional Environment (MATIE) Final Report: Phase I. (AF HRL TR) Brooks Air Force Base, Texas: AFHRL.

11. Larkin, J. H. "Expert and Novice Performance in Solving Phyics Problems," In Science, 208.

12. Park, O., Perez, R. S., Seidel R. J. (1987). "Intelligent CAI: Old Wine In New Bottles, or A New Vintage?," In G. P. Kearsley (Ed), Artificial Intelligence & Instruction: Applications and Methods. Addison-Wesley Publishing Co., 1987.

13. Retelle, John and Kaul, Mike. "The Pilot's Associate--Aerospace Application of Artificial Intelligence." Signal, June 1986.

14. Rich, Elaine. Artificial Intelligence. McGraw-Hill, New York New York, 1983.

15. Waterman, D.A. A Guide to Expert Systems. Reading, MA: Addison Wesley, 1986.

16. Woolf, Beverly and Cunningham, Pat. "Multiple Knowledge Sources in Intelligent Teaching Systems." IEEE Expert. Vol. 2 Num. 2, Summer 1987. The Computer Society of the IEEE.

ABOUT THE AUTHORS

MS. RUTH HAWKINS is a Lead Engineer-Technology with Douglas Aircraft Training Systems and Services/Computer Based Training Systems Division. She is responsible for internal programs including CBT courseware development, internal CBT system implementation, and internal research and development in the area of instructional technology. During her seven years with McDonnell Douglas, she has participated in several aircrew and aircraft maintenance training CBT projects. She received her M.A. and completed coursework for her Ph.D in Educational Psychology at the University of Colorado.

MR. ROBERT ENSEY is an Engineer-Technology with Douglas Aircraft Training Systems and Services/Computer Based Training Systems Division. He is currently serving as subject matter expert for the McDonnell Douglas Ada Software Development Curriculum. His AI interests include the application of heuristic search strategies and natural language processing. He received his M.S. in Computer Science at the University of Denver.

MS. SUE KEMNER-RICHARDSON is a Senior Engineer-Technology with Douglas Aircraft Training Systems and Services/Computer-Based Training Systems Division. She has over 11 years experience in instructional systems design and computer-based instruction. She has been a lead designer of major computer-based training systems in the areas of computer-managed instruction (CMI), task analysis, performance evaluations, and an advanced on-the-job training system. She has also been an instructional designer and has designed and developed on-line authoring support for CMI users. Ms. Kemner-Richardson is completing her dissertation for a Ph.D. in Educational Psychology at the University of Colorado.

AI IN MAINTENANCE TRAINING -- SOME TANGIBLE RESULTS

David J. Sykes
L. C. Keskey P. E.
Honeywell Inc.
Training and Control Systems Division
1200 East San Bernardino Road
West Covina, CA 91790

ABSTRACT

This paper describes the implementation of an AI system that can perform the dual role of Job Performance Aid (JPA) and Intelligent Tutor (IT) for use in On-the-Job Training (OJT). It is well known that the best human experts possess a mental model of internal equipment operation and a good trainer will teach this conceptual knowledge as well as the usual diagnostic skills. The Intelligent Tutor portion is aimed at building this mental model through interaction with a simulation of the equipment. The student interface employs high resolution graphics and a mouse. The simulation is a qualitative causal model which is much simpler than a full mathematical model yet retains all the important distinctions between system states. The Job Performance Aid is an Expert System (ES) which is automatically derived from the qualitative simulation model. This is accomplished by using the model to predict the behavior of the equipment and the propagation of effects under all conceivable conditions. The ES rules are then induced from the fault symptom pattern produced by exercising the model. By taking this approach, the ES provides "deep reasoning" as opposed to the "shallow reasoning" often found in an ES based solely on externally observable features.

INTRODUCTION

In recent years Expert Systems have been successfully applied to a wide variety of maintenance problems. There are, however, several limitations of the current generation Expert Systems. First, they do little to improve the level of understanding of underlying principles by the user. Second, the domain of application is very limited and specific such that unusual situations are not easily handled. Third, most current Expert Systems are based on "shallow reasoning" in which observable symptoms are empirically associated with various end results. Consequently, in addition to providing an Expert System as a Job Performance Aid (JPA) it is necessary to improve the level of understanding on the part of the maintenance technician. It has been demonstrated that the real human experts have developed a mental model of the inner workings of the system being diagnosed. This enables them to diagnose problems they have never seen before (de Kleer and Brown, 1983). The mental model does not need to be at the same level of detail as a design engineer would have. All that is necessary is for the troubleshooter to possess a qualitative causal model which enables him to understand how the system elements interact with each other and how effects are propagated.

While diagnostic Expert Systems are very useful to the maintenance process, they do not reduce the need to develop true human experts. Ideally a cooperative problem solving environment should be provided in which both man and computer (ES) contribute according to their respective strengths (Thomas 1985). This goal can be accomplished by creating an OJT environment in which the function of the maintenance support system could vary as shown in Figure 1. Under crisis conditions it would act as an expert JPA, minimizing time to completion. Under conditions of light workload it would function as a off-line Intelligent Tutoring System (ITS) using advanced explanation facilities and tutoring techniques. Under normal conditions it would perform a mix of JPA and ITS which would help to improve the efficiency of OJT. For the maintenance domain, an expert OJT system could 1) improve the ability of the low end performers to the average level, 2) support the high end performers when they are under time pressure by forcing a rational approach, and 3) preserve eroding expertise in a usable form. The intention is not to replace man but to support him by integration of expert system knowledge with his own experience. For advanced JPA, this calls for a mixed initiative mode of operation in which either man or machine can take the lead depending on the particular circumstances. The basic architecture of a system intended for this dual role is illustrated in Figure 2.

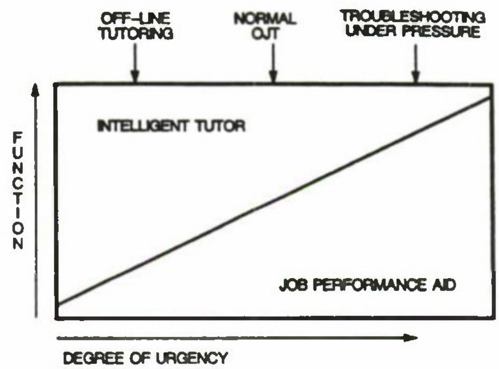


Figure 1. Varying Mix of Tutor and JPA

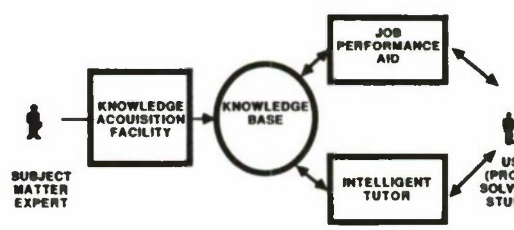


Figure 2. Combined JPA and Tutor

The central thrust of this architecture is the common knowledge base which at a high enough level of abstraction is a simple concept. However, the structure of this knowledge base from an implementation point of view creates a problem. The difficulty is that a different form of knowledge is necessary for instructional purposes than for diagnostic support (JPA) purposes. The first application requires theoretical knowledge of system operation under normal and malfunction conditions.

The second requires knowledge of diagnostic procedures and strategies. The solution to this dual need is a qualitative simulation model of the system in question. The qualitative model (with added tutorial material) can serve as an instructional tool and can also form the source of rules for the diagnostic Expert System.

In this paper the principles of qualitative simulation are described which are then applied to the construction of a specific model for a jet engine oil system. Next, the use of the qualitative model for instructional purposes is described. The procedure for deriving an Expert System from the qualitative model is discussed. Finally, a typical user scenario is illustrated.

QUALITATIVE SIMULATION

Qualitative Simulation is a relatively new technique (Kuipers, 1986) and is based on Qualitative Physics (QP). The goals of QP are to identify the distinctions and laws that qualitatively describe the behavior of physical devices without using quantitative methods. Specifically, these goals are:

1. To be far simpler than classical physics and yet retain all the important distinctions without invoking the mathematics of continuously varying quantities and differential equations.

2. To produce causal accounts of physical mechanism that are easy to understand.

3. To provide the foundations for deep reasoning models for the next generation of Expert Systems

4. To be executed efficiently thus providing a real-time capability.

The principles of QP can be applied to build qualitative models that are suitable abstractions of physical systems for maintenance purposes. The objective is to build a model that represents the causal behavior of a system based on the behavior of its component parts when interconnected in a particular manner. The propagation of effects both normal and abnormal can be studied and used for maintainability oriented problems in design, training, and troubleshooting.

Each state variable of Qualitative Simulation is represented by a small finite number of values known as landmark values. Landmark values can be numeric or symbolic such as low, medium, high, or present, absent or on, off, etc. In maintenance applications many landmark values are binary. The choice of landmark values depends on the criticality of the operational values of the actual variable. Rates of change of landmark variables are represented simply as +, 0, -, to mean increasing, steady and decreasing, respectively. A significant event as in the case of a change in a landmark value is represented by a distinguished time point t. A qualitative simulation has a finite number of distinguished time points. Any qualitative function f can therefore be described in terms of landmark values, derivations, and the finite set of timepoints.

The system can then be described by a series of qualitative functions and a number of constraints. A constraint is used to determine what happens when a landmark value is reached or a certain combination of landmark values of different functions occurs. In other words, the rules of behavior are expressed in terms of constraints. Using this approach, the possible behaviors of a system can be predicted from the initial conditions and the constraints. The behavioral description may then be used to explain a set of observations or the way a system operates.

Once a qualitative model has been developed, it can be used effectively for training maintenance technicians in the following topic areas, or levels of understanding and expertise.

- o Physical structure and connection
- o System functionality
- o Fault detection and diagnosis
- o Optimal troubleshooting strategies

In addition to forming the basis of an instructional device, the qualitative model can be used to derive a diagnostic ES by causal reasoning. The model thus serves as the common knowledge base in Figure 2.

Problem Definition

The size and complexity of this prototype was a major concern in the early project planning phase. It was decided that for a prototype system, a bounded, well-defined, problem of about 50 rules would be the optimum size. It was also decided that a more generic type of system such as an engine subsystem would be a more representative example. Therefore, a simplified version of an aircraft engine oil lubrication system was selected as the subject for this ES-Simulation prototype development project. It was named JEDI for Jet Engine Diagnostic.

The Simulation Model consists of nine components (i.e., objects). Included in the model are the oil tank, strainer, pump, filter, oil air cooler, cold start bypass valve, gear box, oil pressure transmitter, and the oil pressure gauge. This is a simplified version of the actual F-100 Jet Engine Oil Lubrication System. These objects are connected as shown in Figure 3.

Model Implementation

An object oriented programming approach was taken with each of the nine components of the oil system implemented as an object. The objects communicate with each other by exchanging attribute values. Each object is only aware of its immediate neighbors but the effects of a change in one object can be propagated around the network. In Figure 4 a digraph representation shows the exchange of attribute values between adjacent nodes. Note that two extra components have been added (the branch and the join) to cope with modification of attribute values at these junctions. The attributes used between the components are as follows.

- s, oil supply
- p, oil pressure
- f, oil flow
- e, oil temperature
- a, oil foam
- r, resistance to flow

The first five attributes are passed in the direction of oil flow whereas the resistance is transferred in the reverse direction. Thus a change in resistance in the gear box (perhaps due to a change in temperature) will be passed back to the pump where it will affect delivery pressure.

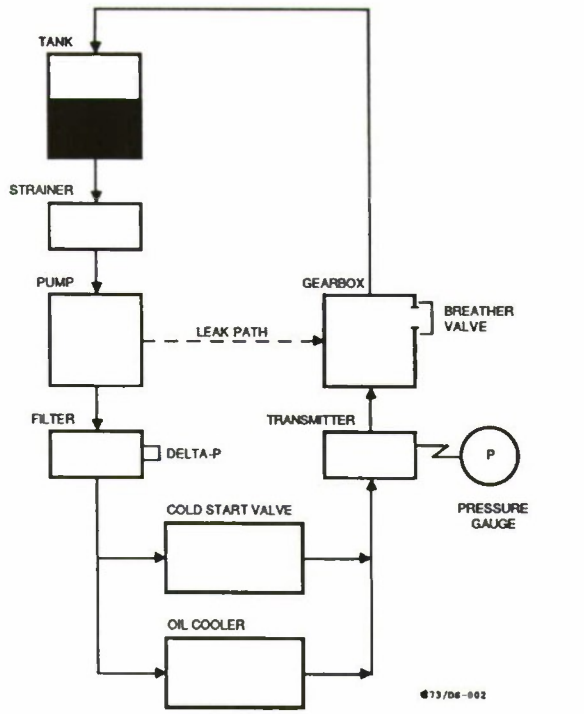


Figure 3. Basic Oil System Diagram

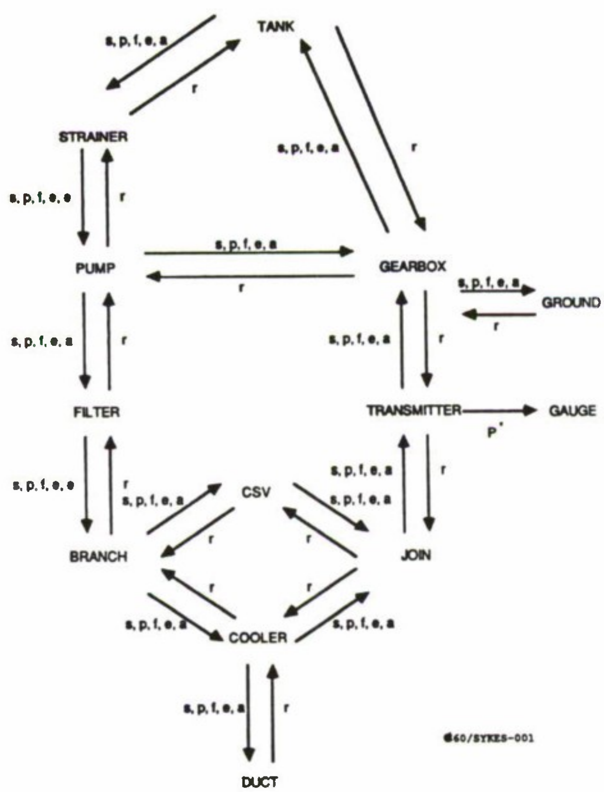


Figure 4. Directed Graph

The behavior of each component in the system is represented by a set of rules. The values of the output attributes are a function of the input attribute values and the current state of the object. For example, consider the pump shown in Figure 5. The rpm is set externally by the model user as is the state which is normal or malfunctioning. One of several rules that define the pump's behavior is as follows.

If the pump state is normal
   and oil supply is present
   and the rpm is idle
   and the resistance is low
Then the output pressure is low

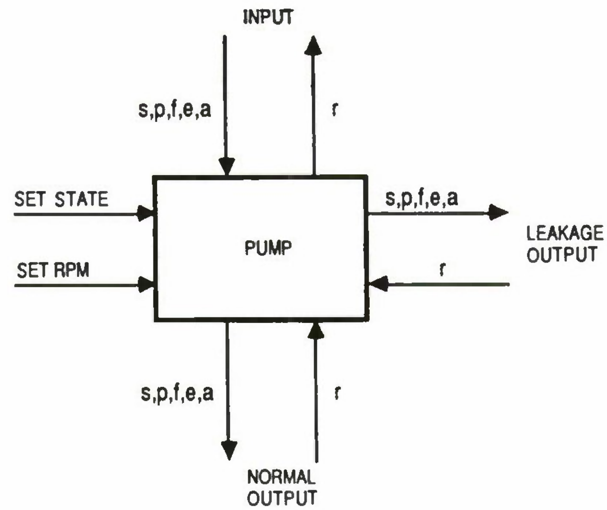


Figure 5. Pump Interface

Each component has its own independent rule set and when connected as shown in Figure 4 the model behaves in a similar manner to the real system.

Results of the Simulation

By running the simulation model several times with a different fault inserted each time, a pattern of attribute values is obtained for each case. For diagnostic purposes this can be used in reverse such that the given pattern of attribute values infers a specific malfunction. Table 1 shows the symptom/malfunction matrix resulting from the simulation. Tables 2 and 3 define the attribute values X and the malfunctions Y inserted at each run. Note that not all attributes are included; e.g., resistance and flow. This is a good example of underlying mechanisms not externally visible which separates deep reasoning from shallow reasoning. The results in Table 1 were then used to derive a diagnostic ES for the oil system.

Generation of the ES

The diagnostic procedure starts with the recognition of an initial symptom,usually zero oil pressure or abnormally low oil pressure. Following this a series of tests is performed to determine further symptoms; i.e., other attribute values. The sequence of tests is based on the value of each test defined as follows.

$$\text{Value of Test} = \frac{\text{Information Gained}}{\text{Cost of Test}}$$

Table 1. Symptom Malfunction Relation

X1	X2	X3	X4	X5	X6	X7	X8	X9	Y
L	2	1	0	0	1	0	0	0	0
H	4	1	0	0	1	0	0	0	0
*	0	1	0	0	1	0	0	0	1
0	0	0	0	0	0	0	0	1	2
L	1	1	0	1	1	0	0	0	3
H	3	1	0	1	1	0	0	0	3
+	0	1	0	0	0	0	0	0	4
+	0	0	0	0	0	1	0	0	5
+	0	1	0	0	1	0	0	0	6
+	0	1	0	0	1	0	1	0	7
L	1	1	0	0	1	0	0	0	8
H	3	1	0	0	1	0	0	0	8
*	0	1	1	0	1	0	0	0	9
L	1	1	1	0	1	0	0	0	10
H	3	1	1	0	1	0	0	0	10

* = any value of X1
\+ = L or H values of X1

Table 2. Attributes and Eligible Values

X	Attribute	Eligible Values	
X1	rpm	0	zero
		L	idle
		H	mil/max
X2	indicated oil pressure (p')	L	idle
		1	low at idle
		2	normal at idle
		3	low at mil
		4	normal at mil
X3	oil supply	0	no oil in tank
		1	some oil in tank
X4	foaming oil	1	no foam
		1	foam present
X5	oil temperature	0	warm (normal)
		1	hot
X6	breather pressure	0	no pressure
		1	normal pressure
X7	oil in the ducts	0	no oil in ducts
		1	oil in ducts
X8	delta-p	0	not extended
		1	extended
X9	oil on the ground	1	oil on ground
		0	no oil on ground

Table 3. Malfunctions

Y Value	Malfunction
0	no malfunction
1	sheared pump shaft
2	warm pump gears
3	cold start valve stuck open
4	breather valve stuck closed
5	ruptured oil air cooler
6	clogged strainer
7	clogged filter
8	loose shut-off value
9	pressure transducer total failure
10	pressure transducer partial failure

The information gained is a function of the relative likelihood of the possible malfunctions that could cause the symptom. The cost of the test is simply the time taken to establish if a particular symptom is present or not. By using the data in Table 1 in conjunction with data on values of the various tests (not shown here), the decision tree in Figure 6 was obtained. This represents the optimum sequence of tests to diagnose problems in the oil system. This was then converted to an ES by translating into a query/response system using actual names as shown in Figure 7. This now forms the JPA portion of Figure 2. The goal of deriving an ES from a description of the system operation by deep reasoning has been achieved.

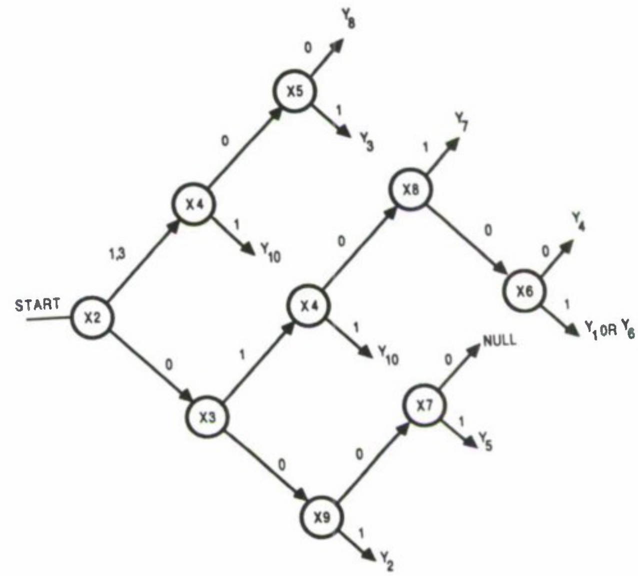


Figure 6. Decision Tree

<u>System Operation</u>

The interface with the user (student and/or problem solver) is shown in Figure 8. Animation using computer graphics is used to display the behavior of the oil system. The user graphics interface also contains a simulated throttle and engine RPM gauge, and an engine start switch (used to activate the simulation model).

A typical simulation model scenario would be to select one of ten possible malfunctions from the menu and "start" the engine. The model receives the malfunction state information from the interface handler, the results or effects of that condition are propagated around the model to all objects, until a new steady state condition is reached.

The user may inspect each object to determine the status of key parameters, such as oil pressure, oil flow, or oil temperature. The student may also inspect the condition or status of any component, such as the pump, or the filter, for the existence of a malfunction. A menu appears on the screen, from which the system user can request component status to be displayed in the component state window.

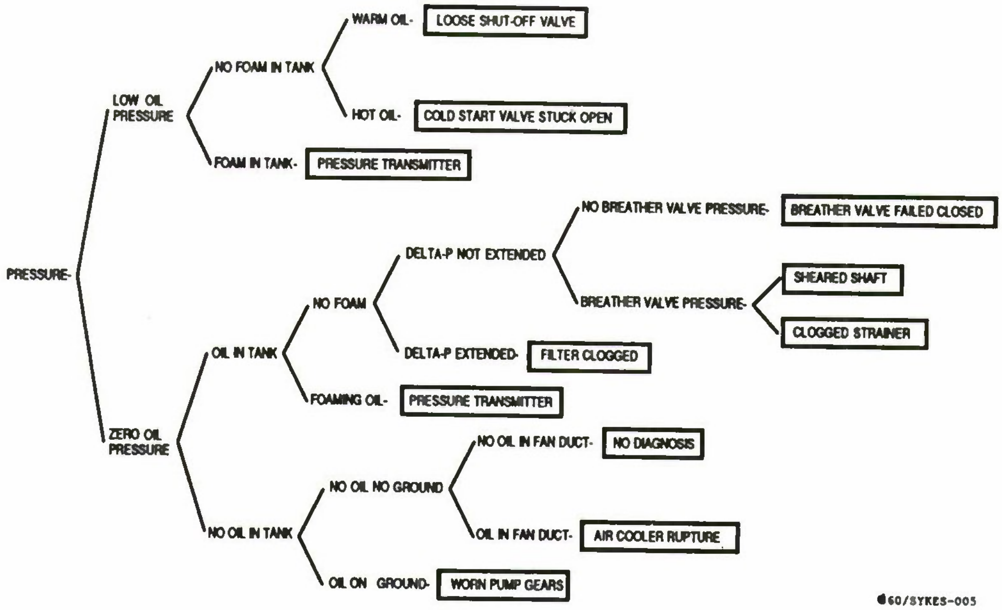


Figure 7. Diagnostic Expert System

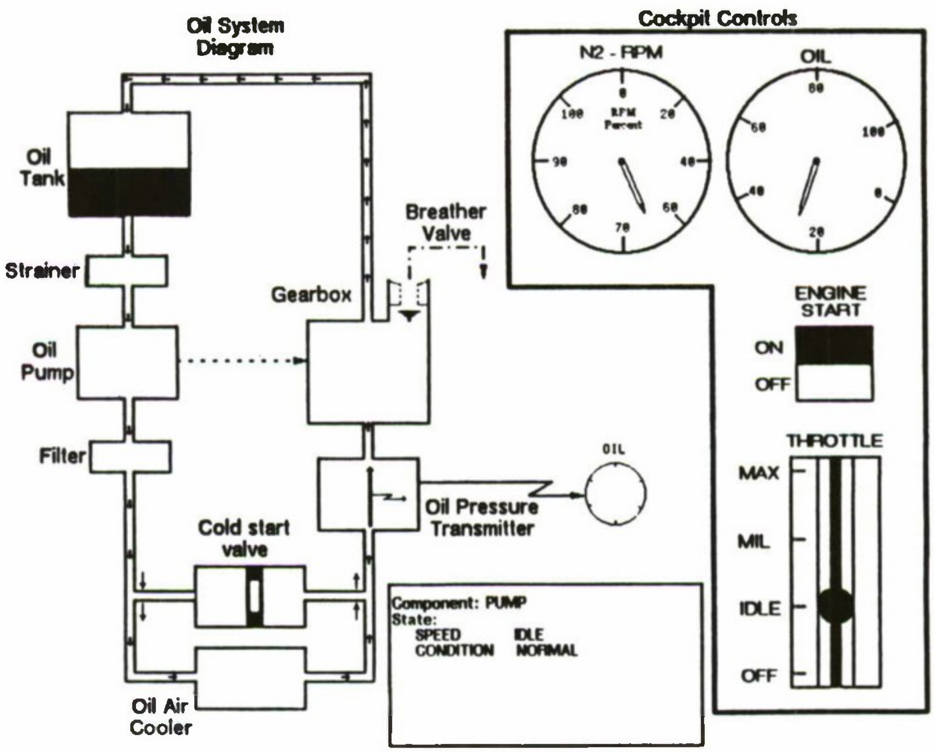


Figure 8. User Interface

Troubleshooting and Fault Isolation

The student may elect to perform his own troubleshooting and fault diagnosis, by examining the various components, until he isolates the fault. Or, the student may select the Expert System option on the Main Menu. In this case, the Expert System guides the student, with a query/response sequence, through an optimum fault isolation and troubleshooting procedure. The Expert System requests certain component status information from the student, based on the student's earlier responses. The student obtains this information by examining the appropriate component in the simulation model's schematic diagram.

In this way, the student may operate in either a practice or test mode. In a practice mode, the student selects the malfunction and performs his own troubleshooting or requests the guidance of the expert system. In a test model, the student can allow the system to select the malfunction, and then perform his own fault isolation troubleshooting procedure.

System Development Environment

The JEDI system was developed on a Xerox 1108 AI workstation. The system configuration included 3.5 Mb, RAM, 40 Mb hard disk, and a 17" graphics monitor (1024 x 808), and a keyboard and mouse. The system was written in Interlisp D, a programming environment based on the lisp programming language. It is widely used within the artificial intelligence community, where it has been used to develop a variety of applications, such as MYCIN (Teitelman and Masinter, 1981). The selection of this programming environment proved very helpful in the actual coding and development phase. The JEDI system including the simulation model, expert system, and interactive graphics interface was completed in a total of ten man days. This included coding, testing, review, changes, and edits.

Recommendations and Future Plans

Following successful demonstration of the prototype JEDI system, the following enhancements and extensions were recommended. First, additional explanation should be added to the JPA Expert System. This feature should not only explain why a certain conclusion was reached, but also explain the significance of the requests for further information as the tree is traversed. When the fault diagnosis is reached, the prescribed corrective action should be requested from the student. This can be enhanced by the use of a video disk system that could also display video/audio segments depicting replace or repair activity. Another recommendation is to add an Intelligent Tutor (which is another form of Expert System) that embodies the skill of a teacher as opposed to the skill of a problems solver.

For the Intelligent Tutor to be effective, it is also necessary to include a Student Model which reflects the student's knowledge and learning abilities. Another area for future work is the true integration of the knowledge bases for JPA and OJT as was shown in Figure 2. This is a nontrivial task. In addition, the scope of the problem should be expanded somewhat so as to represent a real-world practical situation.

ACKNOWLEDGMENTS

The authors would like to thank Horace Barnes of Honeywell T&CSD, David Baum and Chad Thomas of Honeywell Systems and Research Center, Deborah Keifer and Joshua Stern of Xerox AIS, and Jim Wogulis of UCI for their contributions to the JEDI project.

REFERENCES

de Kleer, J. and J. S. Brown, 1983, "Assumptions and Ambiguities in Mechanistic Mental Models." In D. Gentner and A. L. Stevens (Eds.), Mental Models.

Hollan, J. D., E. L. Hutchings, and L. Weitzman, 1984, "STEAMER: An Interactive Inspectable Simulation-Based Training System." The AI Magazine: Summer 1984, 15-27.

Kuipers, B., 1986, Qualitative Simulation," Artificial Intelligence 29:1, 289-338.

O'Keefe, R., 1986, "Simulation and Expert Systems," SIMULATION 46:1, 10-16.

Richardson, J. J., 1985, Artificial Intelligence in Maintenance, Noyes Publications, Park Ridge, N. J.

Stefik, M. and D. G. Bobrow, 1985, "Object Oriented Programming,; Themes and Variations." The AI Magazine, Winter 1985: 40-62.

Teitelman, W. and L. Masinter, 1981, "The Interlisp Programming Environment." IEEE Computer 14:4 (April): 25-34.

Thomas, C. E., D. J. Sykes, A. Scsigulinsky, 1985, "The Impact of Artificial Intelligence on Maintenance Training," in Proceedings of the 7th Interservice/Industry Training Equipment Conference (Orlando, FL, Nov 19-21): 24-40.

Towne, D. M., 1986, "The Generalized Maintenance Trainer: Evolution and Revolution." In W. B. Rouse (Ed.) Advances in Man-Machine Systems Research: Vol. 3, JAI Press, Greenwich, CT.

ABOUT THE AUTHORS

MR. DAVID J. SYKES is a Project Staff Engineer with the Training and Control Systems Division of Honeywell in West Covina, CA. His responsibilities include development of AI programs for training systems and advanced visual simulation systems. Most recently, Mr. Sykes was the Knowledge Engineer on the AI/OJT prototype described in this paper. Prior to this, he was with Honeywell Information Systems in Phoenix and Boston where he designed distributed processing systems and computer networks. Mr. Sykes has published 15 papers on AI, computer networks and simulation of distributed systems. He holds a BS in Electrical Engineering, a MS in Computer Science and is currently completing his Ph.D. in Industrial Engineering at Arizona State University.

MR. LEE KESKEY is a Project Staff Engineer at Honeywell Training and Control Systems Division, Maintenance Trainer Department. Mr. Keskey has extensive experience in design and management level positions in the development of real-time computer based systems. Mr. Keskey is presently involved in the introduction of new technology into the maintenance trainer product line, with emphasis on artificial intelligence and interactive video disc based applications. Most recently, Mr. Keskey directed the successful development of an AI-simulation based trainer prototype to demonstrate the application of Artificial Intelligence and Expert Systems to maintenance training. Mr. Keskey has a BSEE from California State Polytechnic University, an MSEE from the University of Southern California, and an MBA from Pepperdine University.

LOW COST PERSONAL COMPUTER
RIFLE MARKSMANSHIP EXPERT TRAINER (MET)

Albert H. Marshall
Robert T. McCormack
Edward J. Purvis
Ronald S. Wolff
Naval Training Systems Center
Code 731
Orlando, FL 32813-7100

ABSTRACT

The Naval Training Systems Center has developed a low cost marksmanship expert trainer, MET, that allows low cost marksmanship training without an instructor, real weapon or rifle range. The system is safe and does not use costly ammunition. As part of this program, a special long range light pen was developed. The U. S. Navy is currently contemplating the use of this system to teach marksmanship in the Navy's Recruit Training Centers. Teaching marksmanship has required live rounds, special ranges, and a large number of instructors. At present, Navy investment in real estate in close proximity to recruit training centers to construct rifle ranges would be difficult. Also, a large number of experienced instructors would be needed and the high cost of live rounds will add greatly to the Navy's training budget. This paper describes the MET system and the technology applied to this new rifle marksmanship training device. An expert system has been developed to alleviate both the cost and shortage of instructors. The expert trainer is controlled by a personal computer, the Zenith 248. The MET collects real-time shooter performance data or facts, and then executes rules that analyze the trainee performance. Trainee feedback is provided on the computer monitor and by a computer generated voice. The feedback describes the source of shooting errors including improper sight picture, poor shooting position, incorrect trigger squeeze, and incorrect breath control. Through detailed guidance, the novice is able to transition to marksman.

INTRODUCTION

To defend Naval bases and ships against increased security threats, it is imperative that sailors be trained in the basics of marksmanship. In order to teach marksmanship with live rounds, special ranges, weapons, and instructors are required. To do this, the costs and resources requirements are prohibitive. In the case of recruit training, providing such a capability requires substantial resources. The Navy lacks the real estate in proximity to recruit training centers to construct rifle ranges and the shortage of experienced instructors, the high costs of operating ranges and the attendant safety considerations compound the difficulties. The development of a new low-cost transportable training device configured as an expert system for rifle marksmanship training will increase the Navy's capability for providing this instruction.

The MET instructs rifle marksmanship without an instructor, real weapon or rifle range. The expert system is controlled by an inexpensive personal computer, the Zenith 248. This PC is the standard Navy computer and is available through the supply system. The MET system consists of the following components: long range light pen, Zenith 248, color monitor, computer speech board, analog and digital I/O board, and force sensing resistors. MET systems will be networked so that an instructor can provide special help if deemed necessary by the expert system. Networking will permit one instructor to handle as many as eight students.

The MET is based on the four fundamentals of shooting: (1) assume a steady position, (2) put the front sight post on the target, (3) stop breathing and, (4) squeeze the trigger. Sensors attached to the trainee and the weapon measure all of these parameters.

A long range light pen is attached to the M-14 rifle and targets are displayed on the Zenith 248 monitor. The light pen is utilized to determine hits on the target and tracking steadiness. A breath sensor is placed around the trainees diaphragm to determine if he held his breath prior to firing the weapon. A force sensing resistor is utilized on the trigger to determine how the trainee squeezed the trigger.

The trainee can select via a menu, controlled by pointing the weapon at the monitor, which target or training scenario he desires. Feedback is provided by computer generated voice and monitor graphics. Bang and recoil of the weapon are also simulated.

The instructor station will allow the instructor to view the progress of up to eight students. MET will inform the instructor of any student needing special instruction. A photograph of the system is shown in Figure 1.

MET EXPERT SYSTEM PARAMETERS

MET uses the following light pen and sensor derived data to coach the student using the computer generated voice and monitor graphics:

Shot Location - X and Y light pen coordinates
(X = 0 to 639)
(Y = 0 to 199)

Tracking Data - X and Y light pen coordinate data recorded at a 60 Hz rate. A circular buffer of 30 coordinate locations in X and Y is constantly updated prior to trigger pull. These data represent one half second of tracking data prior to trigger pull, and is a function of steadiness.

Trigger Sensor Data - Data from a force sensing resistor is converted and stored in a circular buffer at 60Hz. Data representing one quarter second before trigger pull is analyzed to determine force vs time characteristics. This allows determination of proper or improper trigger squeeze.

Breath Sensor Data - Data from a strain gauge located on a breath sensor belt is converted and stored in a circular buffer at 60 Hz. These data represent breath data one second prior to trigger pull and are utilized to determine if the shooter was inhaling or exhaling at the time of trigger pull.

Using the above light pen and sensor data, mathematical functions are calculated for use by the expert system, to analyze shooter data and provide feedback to the trainee using a computer generated voice or graphics on the monitor.

Presently, the trainee fires ten (10) shots which is a minimum significant statistical sample. Using the data discussed above, the following functions are calculated.

Shot Location Data (10 Shots)

X (mean), Y (mean)
X (standard deviation), Y (standard deviation)
Diameter of shot group

Tracking Data (30 readings - 0.5 seconds prior to trigger pull)

X track (standard deviation)
Y track (standard deviation)

Mean X track (standard deviation) for 10 shots
Mean Y track (standard deviation) for 10 shots

Trigger Sensor Data (10 shots)

Mean and standard deviation of trigger measurement for 10 shots

Breath Sensor Data (10 shots)

Difference in breath sensor output over a one second period prior to trigger pull

Other sensors i.e., rifle butt pressure sensor will be added later to enhance the expert system, if deemed necessary.

Using the above data, the MET expert system generates the appropriate feedback to the trainee. Figure 2 is the MET decision flow chart for the expert system. Much of the knowledge base used to formulate the expert system was derived from a PMTRADE sponsored project that developed an artificial intelligence test bed.

MET SYSTEM CONFIGURATION

The MET system consists of the following components: (Figure 3)

Long Range Light Pen - NTSC Design
Zenith Data System ZVM-1380 EGA-RGB Color Monitor
Zenith ZWX-248-62 Microcomputer and Inteface boards:

MetraByte DASH-08 A/D Converter Board
FTG PXL-350 Hi-Res Light Pen Board
Antex VP-600 Computer Voice Board
Sound System and Head Set
Breath Sensor-NTSC Design
Trigger Jerk Sensor-NTSC Design
Recoil Simulator-NTSC Design

Targets are displayed on the color monitor. A long range light pen is attached to the rifle barrel and is used to determine where the student is sighted on the screen as well as rifle movements prior to firing the weapon. The light pen senses light emitted from a small point on the monitor as the point is scanned across the face of the monitor to create the target display. The light pen has an optical system that limits the amount of the screen viewed to a very small area. When the computer scanned dot is sensed by the light pen optical detector, it sends a pulse to the Zenith computer telling it to read the values of X and Y counters. The counters are controlled by the horizontal and vertical sync from the monitor. The value of the counters are used to determine the lightpen's location at a 60 Hertz rate.

The light pen functions at ranges up to 20 feet from the computer monitor screen, which displays the target and or computer feedback data. The light pen determines where the round would impact on the target and how the trainee is tracking the target, with the simulated weapon.

The light pen uses a small two lens element optical telescope to limit the field of view of the light pen. A photodiode detector with an eye-response optical filter is used to detect the scanned light spot. A transimpedance amplifier converts the light pen current to a voltage that is then amplified. Several electronic filters are used to eliminate power ripple and interfering light sources. The filter has a cut in frequency of 14 KHz. The amplified signal is threshold detected using a voltage comparator. The voltage comparator output pulse is sent to the PXL-350 Hi-Res Light Pen Board. When the light pen detects light from the target monitor, it sets a video latch which freezes the counters. After the Zenith software reads the counters, it clears the video latch and re-enables the X-axis and Y-axis counters. The counters determine the X and Y location the light pen is pointing on the target monitor screen. A separate 12-bit counter is used for both the X and Y coordinates. The X - axis counter is clocked with a 30 Mhz signal that provides single pixel resolution on the monitor screen. The X-axis counter is reset to 0 by horizontal sync pulses, (horizontal retrace). The Y-axis counter counts horizontal sync pulses, which is equivalent to counting the number of lines. The Y-axis counters are reset to 0 on vertical sync pulses (vertical retrace). When light is detected by the light pen, the counter gates interrupt the clock and reset signals to the X and Y counters. This freezes the counter state when the video latch is set, by the light pen. The counters are frozen with the X and Y coordinate data until the system reads the counters and clears the video latch. Figure 4 is a block diagram of the light pen and light pen board. Figure 5 is a photograph of the light pen.

The Zenith 248 personal computer was chosen because it is a Navy/DoD standard computer and is available in the supply system. The Zenith utilizes an Intel 80286 microprocessor and is IBM PC/AT compatible. The operating system is MS-DOS V3.2.

The display is a ZVM-1380 RGB color monitor and is EGA compatible.

A MetraByte 8 channel high speed A/D converter is used to collect data from the breath sensor and trigger jerk sensor. It is also utilized to control the recoil simulator. The A/D has 12 bit resolution.

A force sensing resistor (FSR) is attached to the trigger to measure if the trainee is squeezing or jerking the trigger. The FSR is a new type of thick film electronic component. The resistivity across the device drops in a non-linear fashion as the applied force, perpendicular to the sensor, is increased. The FSR consists of two parts sandwiched together. The first part is a special conductive polymer. The second part is a conductive finger arrangement. The two parts are formed by silk-screening the appropriate materials onto mylar sheets of various thickness, size, and shape.

A very small Shunt-Mode style FSR was constructed on a 5 mil mylar and placed on the surface of the trigger. A non-linear amplifier utilizing the characteristics of an ordinary diode was used to linearize the sensor such that useful analog data could be obtained. The signal was then conditioned for input to the A/D converter.

A breath sensor is strapped around the trainees chest to determine if he has frozen his breath, prior to firing the weapon. Two strain gauges are used in a bridge configuration to double the overall sensitivity. The sensors are mounted on a flexible material that will flex according to the trainees breathing pattern. The very small electrical signal, generated from the bridge arrangement, is then amplified by a Differential Instrumentation amplifier. The signal is further filtered and conditioned for input to the A/D converter.

Recoil is simulated by pulling the weapon from the rear with a flexible cable. The recoil simulator consists of an electric motor, electromagnetic clutch, and flywheel. When the rifle is fired, the clutch is energized for milliseconds engaging the flywheel pulling the cable which is attached to the top of the rifle butt plate. The recoil device will accommodate any firing position.

The trainee wears a head set that is used to both simulate the weapon firing report and provide computer voice feedback. The feedback is individualized based on the trainees progress. Various messages sent to the trainee are shown in Figure 2. Future vocabulary words can easily be added to the trainer.

CONCLUSIONS

The laboratory marksmanship expert trainer has been fabricated and demonstrated with good success. At present, a prototype system is configured with one instructor station and three trainee stations. This system will undergo evaluation testing in August at the Naval Training Systems Center in Orlando, Florida and the results will be reviewed at the conference.

The initial system has been very reliable and all components are low in cost. Using expert systems principles, a trainee can be taught to shoot without a range or live rounds and use a minimum of instructor time.

This application of expert systems is unique in that it uses sensors on a weapon as well as a trainee to teach a psycho-motor skill. Its effectiveness should stimulate the application of expert systems to other psycho-motor training needs.

ABOUT THE AUTHORS

Mr. Albert H. Marshall is a Team Leader/Physicist at the Naval Training Systems Center. He has specialized in developing weapon fire simulator using lasers, electro-optics and microprocessors. He holds twenty one U.S. patents. He has Master's Degrees in both physics and electronics engineering from Brown University and the University of Central Florida.

Mr. Edward J. Purvis is an Electronics Engineer at the Naval Training Systems Center. He has specialized in developing weapon fire simulators using electro-optics, microprocessors, video processors, and graphics. He has a Masters Degree in Digital Electronics from the University of Central Florida.

Mr. Ronald S. Wolff is an Electronics Engineer at the Naval Training Systems Center. He has specialized in developing weapon fire simulators. He has worked with electro-optics and interfacing various non-linear sensors to microprocessors. Mr. Wolff graduated from the University of Central Florida in 1985. He is currently pursuing his Master's Degree in electronics engineering at the University of Central Florida.

Mr. Robert T. McCormack is an Electronics Engineer at the Naval Training Systems Center. Specializing in weapon fire simulators, he has worked on computer program development and sensor interface to microcomputer/microprocessors. Mr. McCormack graduated in 1985 from the University of Central Florida, where he is currently pursuing his Master's Degree in Digital Electronics Engineering.

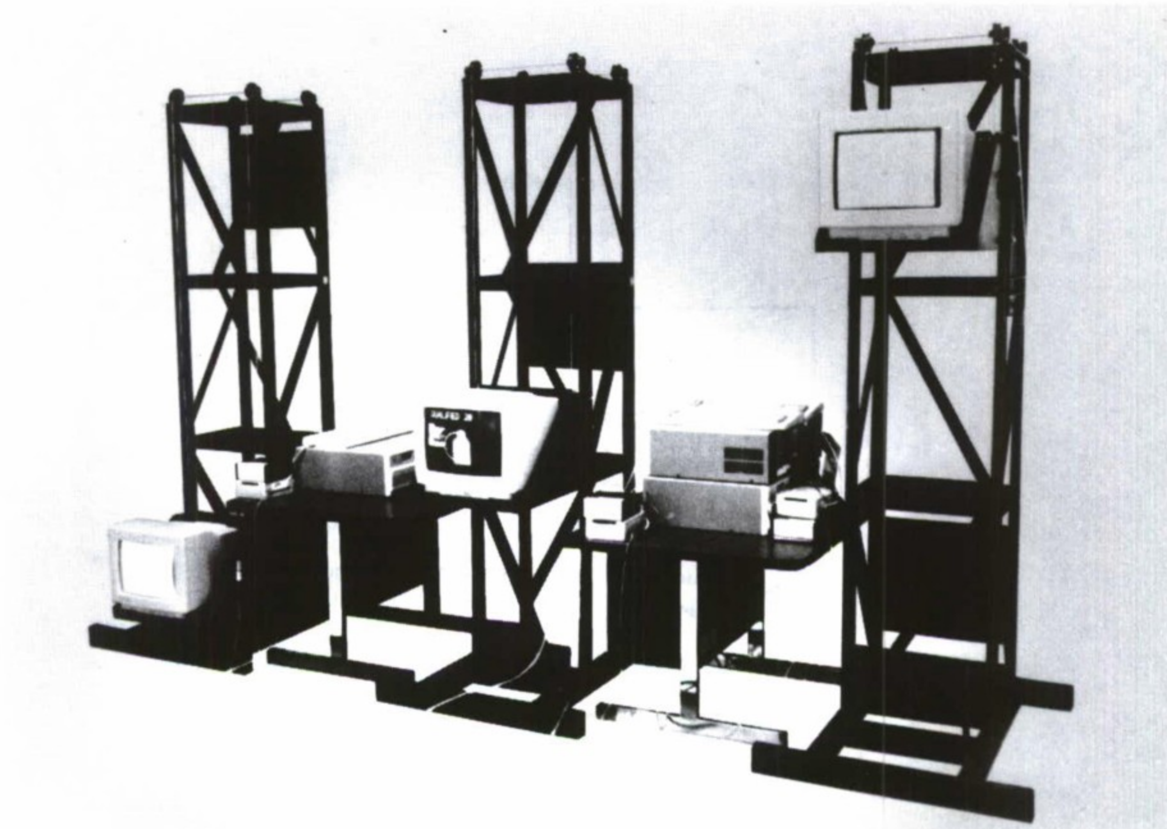


Figure 1A. System (Three (03) Target Monitors and Computer Systems)

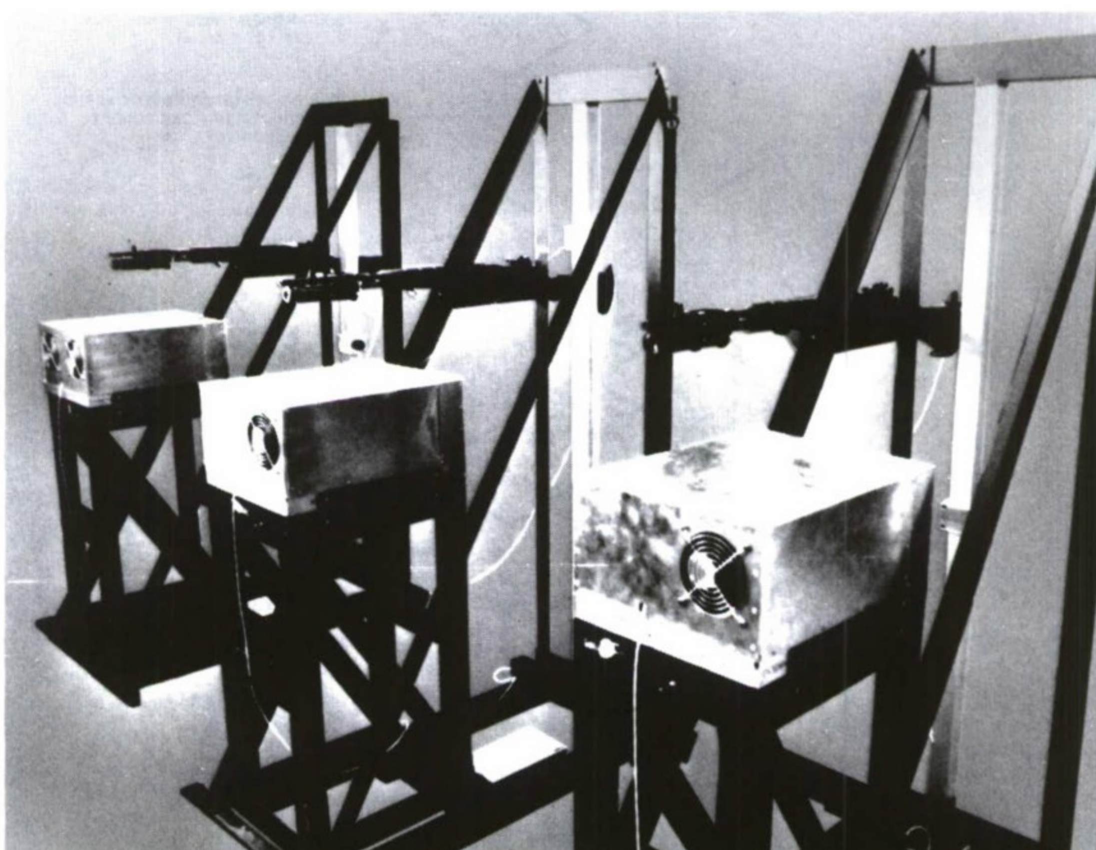


Figure 1B. System (Three (03) Weapons and Recoil Devices)

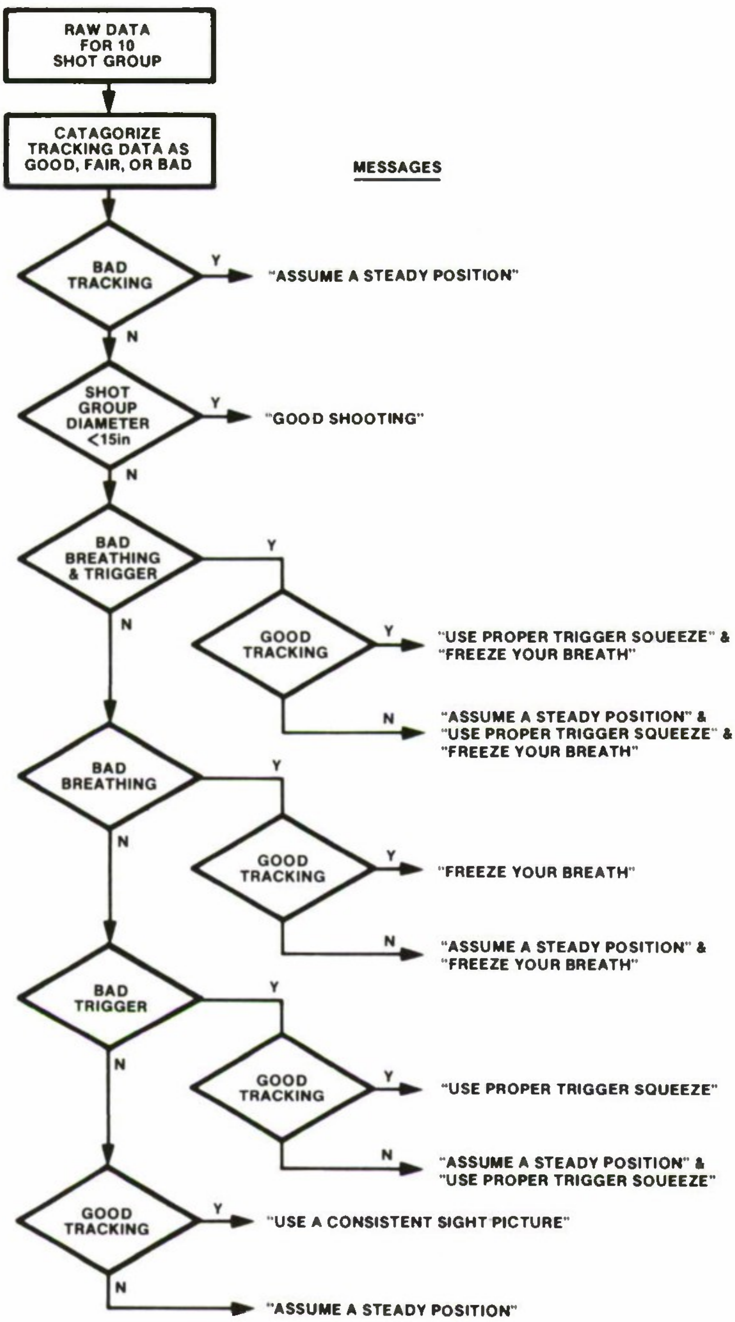


Figure 2. MET Expert System Flow Chart

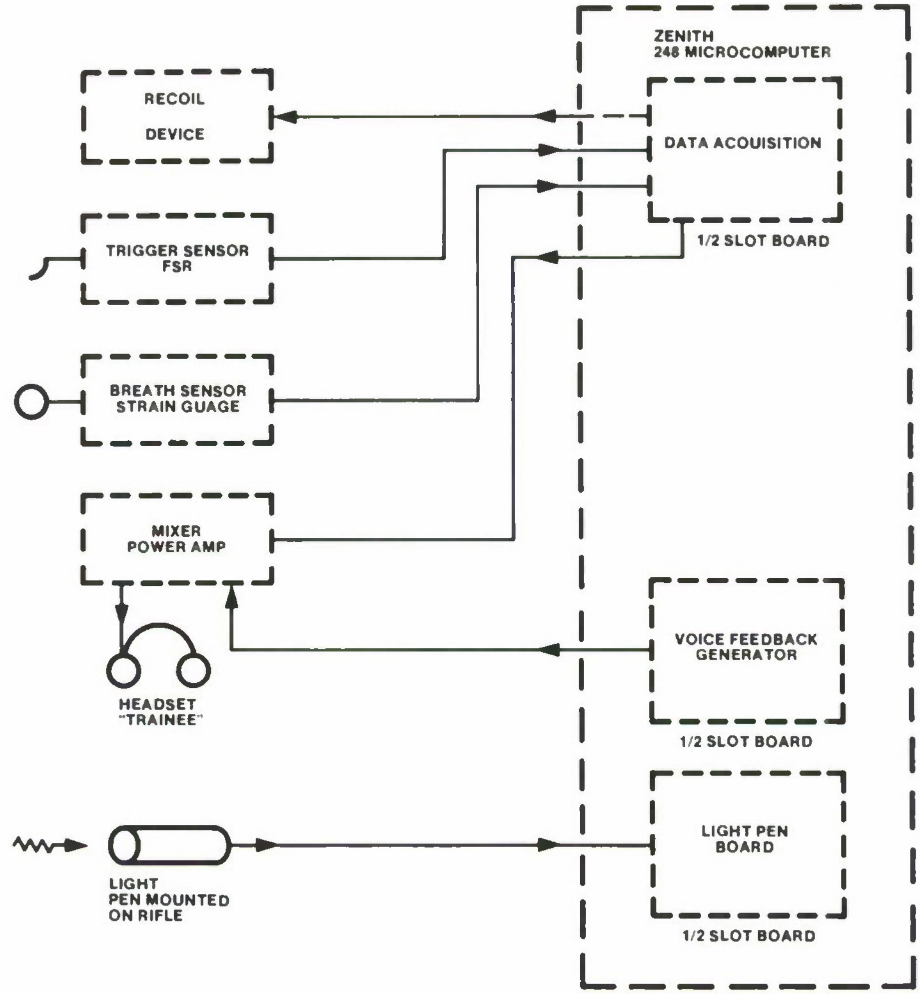


Figure 3. MET System Block Diagram

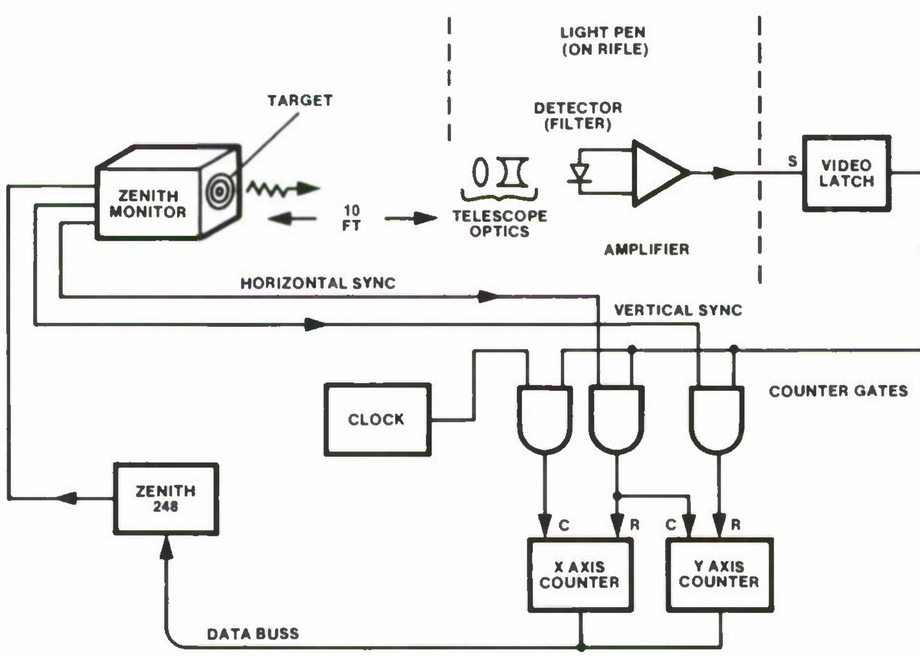


Figure 4. Light Pen and Light Pen Board Block Diagram

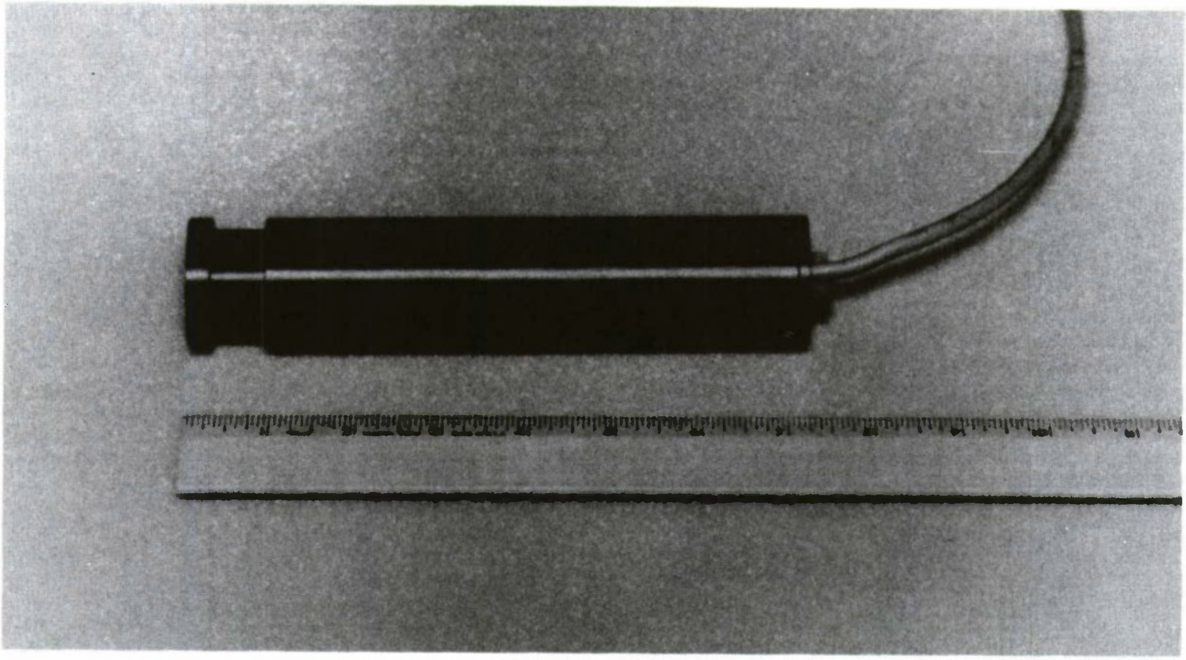


Figure 5. Long Range Light Pen

MARS: A TARGET PROJECTION SYSTEM FOR AIR COMBAT SIMULATORS

Pierre Rapp
THOMSON-CSF Division Simulateurs
Osny, FRANCE

ABSTRACT

The operational experience and technical know how acquired in air combat simulation has led THOMSON-CSF Simulator Division to develop a new simulator projection system. This equipment called "MARS" for Multiple Aircraft Raid Simulation, provides combat pilots with better perception of their targets for multi aircraft training by overcoming the limitations of light valve projectors. In particular, the use of laser and acousto-optical techniques provides high contrast images without halo. The images displayed by MARS are usually ground or air targets, missiles in flight and the sun, but it is also possible to display gunnery effects such as tracers and countermeasure decoys.

INTRODUCTION

Simulation of an air combat visual environment in domes is currently obtained by superimposing on a low resolution background image of land and sky, brighter high resolution images of small objects such as airborne targets and missiles of various sorts which are important to the pilot.

The increasingly improved performance of modern combat aircraft has led THOMSON-CSF to develop the combined MARS-JANUS system to meet all the operational requirements for an air combat simulator for these aircraft, namely:
- full compliance with the pilot's field of view,
- improvement of the terrain/sky image, particularly at low altitudes,
- improvement of the resolution and quality of the projected targets
- reproduction of a multi-target environment.

The JANUS system, which is not the subject of this paper, meets the first two requirements by projecting a dynamic full field image giving the pilot speed, altitude and attitude visual cues.

The MARS system provides a decisive reply to users' insistent requests for improved target image resolution and quality. The outcome of an air combat in a complex multi-threat environment is very closely linked to the rapid and unambiguous identification of the target(s) and the exact evaluation of its behavior.

To achieve this, it is important that the image be bright with high contrast sharp edges which give a good indication of shapes, therefore of the type of aircraft, its armament and its instantaneous attitude.

Existing target projection systems use CRTs or light valve projectors whose low contrast and the presence of a halo around the target limit the quality of the target images by their insufficient contrast. To be perceived, the target images must be brighter than the terrain/sky background. The low contrast of existing projectors (75 to 100:1) gives rise to an objectionable halo around the target image especially when the image is projected against the darker parts of the terrain background. The halo effect increases with projector brilliance, degrading the content of the image itself and reducing the perception of detail in the image.

A good quality image is particularly required when the targets are at distances critical for combat i.e. when the result of the encounter depends on the rapid and reliable identification and evaluation of the targets by the pilot - namely between 1500 meters and 6000 meters.

The MARS target projector considerably improves the luminosity and sharpness of the image in this particular range of distances, thereby giving the pilot the vital target information he needs.

OPERATIONAL REQUIREMENTS

Pilots judge combat simulator target projectors essentially on their capacity to provide good perception of targets at long range. Experience in air combat has shown that there are three main domains:
- short range: targets closer than 1500 meters,
- medium range: targets between 1500 meters and 6000 meters,
- long range: targets beyond 6000 meters.

In the first two domains, pilots require to recognize:
- the type of aircraft and whether it is friendly or hostile,
- its weapons,
- any changes in its configuration such as reheat, airbrakes, variation in wing geometry, etc.

The medium range domain is the critical domain in air combat because it is here that the visual identification can be decisive. All other things being equal, it is the first pilot to see the other who stands the best chance of winning. "He who sees first lives longest". The best projected target quality must therefore be provided in this domain.

At long range, pilots have no special requirement other than an image showing the presence and position of the target. They certainly do not wish to have the detection of the target facilitated by the presence of a halo.

Note: the above considerations are clearly subject to fluctuation depending on weather conditions and the pilot's visual acuity under combat constraints (altitude, fatigue, workload).

Research undertaken by THOMSON-CSF with the cooperation of the CERMA (French Aerospace Medicine Study and Research Laboraroty Center) concerning the mechanism of recognition of small objects and the sensitivity of the human eye, showed that to meet the user's requirements, it was necessary to improve the contrast by removing the halo surrounding the target image (see figure 1) and to improve the brightness and resolution of small images.

These improvements are obtained by replacing the conventional light valve projector by a laser based MARS projector.

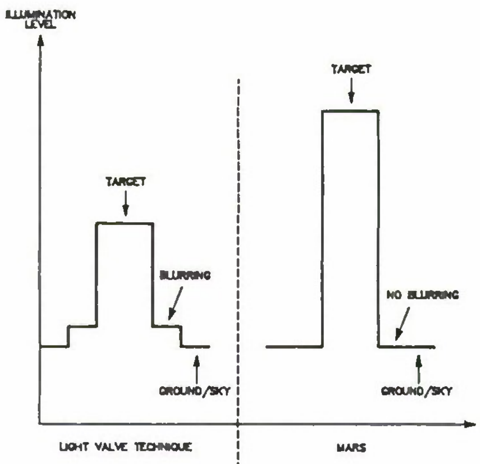


Figure 1

DESCRIPTION OF THE MARS PROJECTOR

The MARS projector consists of the following main subassemblies (Figure 3):

- the projection system itself comprising:
  - . a laser source,
  - . an energy distribution device,
  - . several independent projection heads.
- a dedicated computer system which computes the parameters and commands for the computed image generator (CIG), the video and servo modules.
- digital servo modules driving the projection heads.
- a video module linking the CIG to the projection head.

Projection system

The laser source emits a blue beam and a red beam. The combination of these two beams gives a white which the International Commission on Illumination has recommended for good visual comfort.

Each beam is divided into as many beams as there are projection heads, the light being conducted to the heads via optical fibers. This arrangement has the advantages of simplicity and flexibility and minimizes the space occupied by the system in the dome.

The projection heads consist of the following:

- a bichrome modulator (red and blue),
- an optical mixer combining the red and blue modulated beams,
- an optical attenuator for adjusting the luminosity for distance and gray out,
- an X-Y scanner which traces the image,
- an optical device performing the functions of image transport, zoom and focus.
- an azimuth and elevation deflector with motor driven mirrors.

Acousto-optical modulator assembly

. Principle (Figure 3)

The laser beam to be modulated traverses a lead molybdate crystal which is also traversed by an acoustic wave. The incident beam is transformed into a beam transmitted without modulation and a diffracted beam modulated by the acoustic wave. The diffracted beam contains about 90% of the incident energy. This efficiency is optimized by making the incident angle equal to Bragg's angle.

1 Incident beam
2 $PbMoO_4$ crystal
3 diffracted beam
4 piezo-electric transducer
5 modulating signal
6 transmitted beam

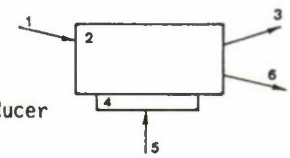


Figure 2

. Construction

Each bichrome projection channel has a modulator with two separate crystals (one for the red beam and one for the blue beam) giving the additional capability of providing a means of continuously varying the amount of blue or red in the composite beam (if required).

At the output of the modulator an optical mixer combines the beams. The resulting white beam is then sent to the scanner.

Scanner assembly

The X-Y scanner specially developed by THOMSON-CSF generates either a TV image or a calligraphic image from the modulated beam. The XY scanning signals are synchronized with the video signals driving the modulators.

Optical assembly

An optical assembly located between the scanner and the deflector provides the functions of image transport, image size (zoom) and focussing. The zoom and focus drive signals are computed in the MARS system computer. The size of the projected target is adjusted by an optical zoom for close distances and by an electronic zoom located in the CIG for distant targets. The optical zoom is selected to meet the customer's requirements for the range of target distances and to provide a resolution compatible with the separating power of military pilot's eyes which happens to be better than the average (1' of arc).

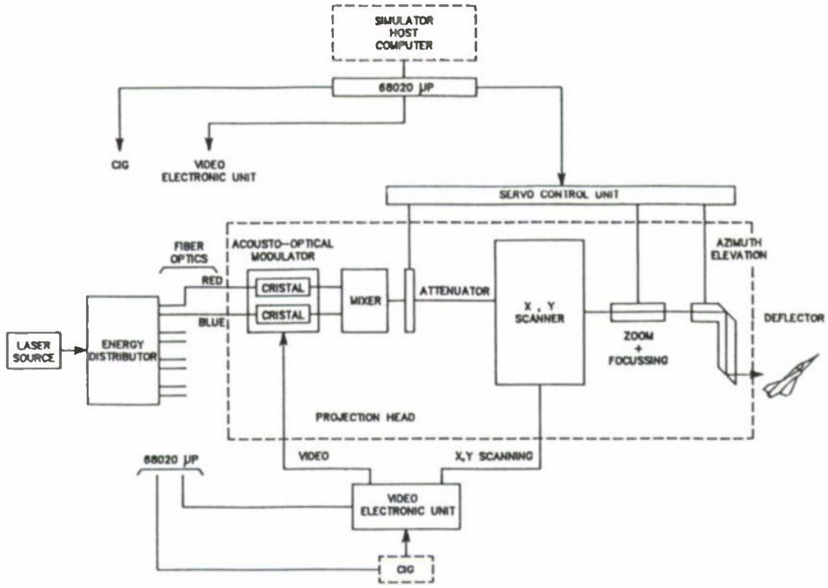


Figure 3 MARS GENERAL OUTLINE

Deflector

The miniaturized deflector is of conventional design and provides unlimited movement in azimuth and elevation.

Dedicated computer

The MARS projection heads are controlled by a microcomputer facility based on a MC68020 microprocessor and an MC68881 floating point coprocessor. The processors interface via a VME bus.

The software loaded in the computer includes support software, application software (operation of the projection heads and computation of rotational motion) and integration and maintenance software.

The computer interfaces with the following:
- the simulator host computer for target and fighter positions and attitudes,
- the image generator for transmission of parameters of the displayed targets,
- the servo modules for transmission of drive commands to the electro-mechanical devices.

Video module

The video module provides the following main functions:
- adaptation of the CIG image to the characteristics of the scanner
- generation of the video signals for the modulator,
- generation of the video synchronized scanning signals for the scanning modules.

Servo module

The following commands are sent to the projection heads:
- the positions of the azimuth and elevation deflector mirrors,
- the zoom lens focal length,
- focussing,
- attenuation.

The deflector azimuth and elevation mirrors are driven by torque motors and have unlimited rotation to allow projection to any part of the dome. Digital servo systems are used, providing better fidelity and maintenance flexibility. The feedback signals are also input to the computer to provide enhanced monitoring of the servo loops.

Safety

In addition to the usual safety precautions applied to laser installations, THOMSON-CSF has paid particular attention to the safety aspect,

providing complete protection to all persons entering a simulator equipped with a MARS system. System operation depends not only on the correct operation of its subsystems (presence of scanning for example) but also on conditions related to the current phase of simulation. For example, in normal operation the projectors will not operate unless the pilot is in his seat with the canopy closed. This prevents the pilot from being exposed to direct rays from the projectors which are out of his normal field of view.

PERFORMANCES

Typical performance data given below illustrate the capacity of the MARS system to project high contrast, high resolution target images. The performances can be tailored to suit the user's specific requirements.

The following assumptions have been taken into account:

- diameter of dome: 8 meters,
- dimensions of the real target: 15 meters (e.g., Mirage 2000)
- pilot's eye at the center of the dome,
- optical zoom chosen to obtain resolution better than the human eye,
- screen gain: 0.9.

Projection head servo speed and accuracy are essential especially when it is necessary to switch the image from one head to another when the projection heads are located on each side of the cockpit. The small size of the heads gives minimum occultation.

static accuracy per axis: better than 1 mrd,
max. angular velocity per axis: 20 rd/s,
max. angular acceleration per axis: 200rd/s²,
separation between mirrors: 60mm,
mean swept diameter: 80mm.

A/C - target range (meters)	150	300	1200	6000	
Zoom	1 <	2 optical	8.4 ><	electronic	>
Definition in lines per image	256	constant	256	variable 51	
Resolution in Min. of arc	1.7	0.85	0.2	constant 0.2	
Luminosity lux ft L	>40 >4	>40 >4	>100 if necessary >10 if necessary		
Contrast			>500		

In addition to the high image quality the innovative design of the MARS system has two outstanding advantages:

- multi-target capability because of the advanced level of miniaturization of the projection heads,
- display of high light level special effects such as afterburners, missile firing flash, IR decoys, tracers, navigation lights using the calligraphic mode.

APPLICATION TO SIMULATORS IN DOMES

The unique features of the MARS system make it exceptionally suitable for use as a multiple target projector in air combat and gunnery simulator domes. The product was designed with sufficient flexibility to enable it to interface with existing installations such as connection to any host computer, adaptation to various simulator cockpits and use of specific CIG.

It is quite possible to arrange dynamic management of the projection heads so that the same head projects either a target or a missile, etc., depending on the tactical situation at the time and/or the direction in which the pilot is looking (area of interest).

An artist impression of a typical MARS/JANUS* installation in a dome is shown in Figure 4. The projection heads are arranged in pairs on the left and right of the cockpit. The switching of the image when the projected image passes from one side of the cockpit to the other is performed out of the HUD field of view.

* JANUS (from the two-faced Roman god) is a system developed by THOMSON-CSF which uses two fisheye projectors to project a terrain/sky image on the front hemisphere and the rear hemisphere simultaneously. The two images provide the pilot with realistic cues of altitude changes, aircraft attitude and ground speed.

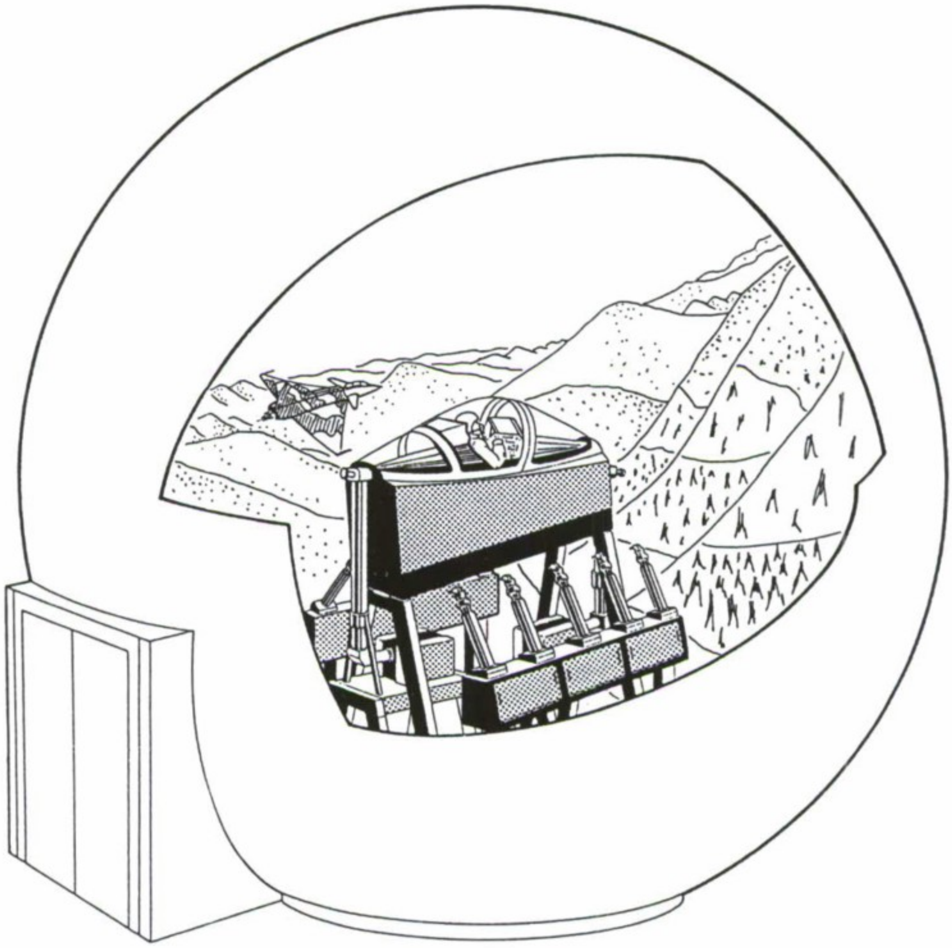


Figure 4

CONCLUSION

The main feature of the MARS system recently developed by THOMSON-CSF is its capacity to project targets having sufficient contrast and resolution to enable the pilot to identify and evaluate targets at medium and long ranges. The system's modularity gives it the capability of projecting a diversity of moving objects ranging from target aircraft to tracer bullets, thereby creating the multi-threat environment that future air combat simulators will require.

REFERENCE

Les fonctions de sensibilité aux contrastes de luminance polychrome chez l'homme (Study of human contrast sensitivity to polychrome luminance).
Thèse Docteur MENU CERMA 5/12/1986.

ABOUT THE AUTHOR

Pierre Rapp is a project Manager in the fighter aircraft simulation department at THOMSON-CSF Simulator Division. He is currently involved in R & D coordination on air combat simulators. He has been working in the simulation field since 1969.

HELICOPTER SHIPBOARD LANDING RESEARCH AT THE VISUAL TECHNOLOGY RESEARCH SIMULATOR

Daniel P. Westra
Essex Corporation
1040 Woodcock Road
Orlando, Florida 32803

Dennis C. Wightman and Joyce J. Madden
Naval Training Systems Center
Orlando, Florida 32813-7100

ABSTRACT

Simulator design and instructional issues for helicopter shipboard landing operations are presently under investigation at the Navy's Visual Technology Research Simulator (VTRS) following the recent installation of a Vertical Take-Off and Landing (VTOL) simulator. Research strategy at VTRS to provide answers for applied training problems has employed economical multifactor experimental design to deal with the many factors which may influence performance and an iterative three phase process to deal with "transfer of training" as the ultimate issue. The first phase of this process consists of performance studies in which the effect of various design features on experienced pilots are examined in the simulator. The second phase consists of in-simulator transfer-of-training experiments in which pilots novice to the task are trained under various simulator configurations and instructional conditions and then tested under a high fidelity simulator configuration. The third phase employs the transfer-of-training experimental paradigm with training in the simulator and testing at an operational site. Currently, the VTRS helicopter shipboard landing research program is in the second phase. This paper presents results from two major performance experiments already completed, and show how the results were used to progress from the first experiment to the second and then to the current in-simulator transfer-of-training experiment, which will also be discussed.

INTRODUCTION

A major focus of the research effort at the Navy's Visual Technology Research Simulator (VTRS) is to experimentally evaluate simulator design options and training procedures for important flight tasks. This research provides guidelines for (1) decision making for flight simulator design options, and (2) the development of instructional procedures to achieve optimal use of simulator training time. Currently, simulator design and instructional feature issues for the helicopter shipboard landing task are under investigation. A program of research is underway which includes performance experiments and transfer-of-training experiments.

RESEARCH STRATEGY

A three-phase research process, combined with principles of economical multifactor experimental design, has been employed to quickly investigate many simulator design and instructional feature issues economically and in a reasonable period of time. The three phase approach has been used previously in VTRS research to determine simulator design requirements for the carrier landing task [10, 7, 12] and for air-to-ground weapons delivery [8, 3, 2]. This research has been partially summarized by Lintern, Wightman and Westra [4]. This process is iterative in nature wherein information obtained at each phase is used in the planning and design of succeeding phases. The first phase of this process consists of performance experiments in which the effects of various design features on experienced pilots are examined in the simulator. The second phase of this process involves in-simulator transfer-of-training experiments which employ the transfer-of-training experimental paradigm. In this phase, pilots novice to the task are trained in the simulator under various simulator configurations and instructional conditions and then tested under a high fidelity simulator configuration. The third phase employs the transfer-of-training experimental paradigm with transfer testing taking place in the field. Currently, the VTRS helicopter shipboard landing research program is in the second phase. This paper will present results from two performance experiments already completed, and show how the results were used to progress from the first experiment to the second and then to the current in-simulator transfer experiment.

Although transfer of training is the bottom line in training research, performance experiments are extremely valuable, indeed necessary, as the first phase of a research program investigating the effect of a large number of factors. First, they serve as a vehicle to develop and validate performance measures and experimental procedures. Second, they serve to "screen" variables for subsequent transfer experiments. Factors that have little or no effect on performance are unlikely to affect transfer and may be excluded from transfer experiments. This assumption may be questioned, but exceptions are difficult to find in the literature. Preselection has always been a part of planning for transfer studies, and in cases where theory and prior data do not offer a useful guide, performance experiments provide a rational basis for factor selection.

Third, the results from performance experiments, particularly in the case of null results, do provide direct information regarding skill maintenance or transition training for experienced pilots. And, finally, by taking advantage of experimental designs which use the same subject across numerous conditions, a great deal of information can be obtained at relatively low cost. This means that even very large-scale multifactor experimental designs can be conducted with only relatively few representative pilots. In contrast, pilots can perform on only one training condition in a transfer-of-training

experiment, so that the conduct of even an in-simulator transfer experiment of any reasonable power requires a great deal of resources. Conducting a field transfer-of-training experiment requires even greater resources, introduces difficult logistic problems, and can reduce experimental control. Thus, it would appear not only prudent and economical to employ this research strategy, but necessary in the case of a many factor problem for which generalizable, applied results are desired. In all three phases of research a holistic experimental philosophy is used as proposed by Simon [6].

RESEARCH PROBLEM

Landing a helicopter on a small ship is a particularly difficult task and that difficulty is accentuated in turbulent seas. Typically, the pilot establishes the aircraft on a descent path about one mile behind the ship and approaches the landing area while reducing speed. The pilot transitions to a hover relative to the ship at a height of approximately 15 feet above the landing deck. A hover is maintained above the touchdown point until the pilot ascertains that the deck is level and stable enough for a safe landing. At that moment, the aircraft is quickly lowered to the landing area and secured to the deck. Communications from the fleet indicate that their present simulator is not satisfactory for teaching the final stages of the helicopter approach and landing.

The Light Airborne Multipurpose System Mark III (LAMPS MK III) integrates an FFG7 frigate and a SH-60B Sea Hawk helicopter to provide an over-the-horizon detection and strike capability for antisubmarine warfare and antiship surveillance and targeting. The system has recently been introduced to the U.S. Navy and it is anticipated that approximately 100 units (i.e., 100 ships and 200 aircraft) will eventually be deployed. The SH-60B cockpit was installed at the VTRS facility so that simulator design and instructional issues for helicopter small ship operations could be examined.

VTRS RESEARCH FACILITY

The simulation at the VTRS facility supporting helicopter training research includes an SH-60B cockpit with all displays and controls that are important for flight control and guidance. These displays and controls function in real time and closely simulate those of the aircraft within the flight regime of the approach and landing. The cockpit is mounted on a fixed base in a 17-foot (5.18m) radius dome. It has a pneumatic g-seat, with buttock, thigh, and back cushions that simulate tactile pressures experienced in flight. Twin 1025-line color projectors are used to provide a 160 degree (H) by 70 degree (V) computer generated image of the outside visual scene. This field of view is set 40 degrees to the left, 120 degrees to the right, 20 degrees above, and 50 degrees below the forward line of sight set for helicopter operations. Maximum scene brightness is approximately 0.2 ft Lamberts (0.685 cd/m^2). Herndon [1] provides a more complete description of the VTRS helicopter simulator.

PERFORMANCE EXPERIMENT I

Eight experienced Navy pilots made approaches and landings on a representation of an FFG7 frigate in the vertical take off and landing (VTOL) simulator at the VTRS. The pilots were from operational squadrons and routinely flew VTOL aircraft in helo/ship operations. The experimental task involved the approach and landing of the simulated SH-60B helicopter to a simulated FFG7 frigate moving forward at 10 knots (18.5 Km/h). The aircraft was initialized at 160 feet (48.4m) altitude on an approach heading 2000 feet (609.3m) behind the ship. The simulated aircraft was initialized at an airspeed of 43 knots (79.9 Km/h) for a descent rate of 128 feet per minute (39.0 m/min) descending approach to the FFG7. The glideslope approach angle was nominally set at 3.5 degrees, although no glideslope indicator was available and pilots were not specifically concerned with maintaining that approach angle. The approach and landing involved a descending and decelerating approach to the ship, transition to hover near the stern, hover over the landing area, and descent to the designated rapid securing device (RSD). A complete description of this experiment and the results was reported in Westra and Lintern [11].

Factors and Levels

Factor and level settings represented those of most interest and were generally chosen to bracket the reasonable range of interest. "High" factor levels were generally set at the highest fidelity attainable under VTRS capabilities. Low level settings represented the most degraded form of the factor likely to be used in an Operational Flight Trainer (OFT) or a level currently being used in a flight trainer. One exception to this was the scene detail factor whose "low" level was little more than an outline of the ship with solid surfaces. This was done as the first step in a process of identifying and isolating specific scene detail effects.

In all cases, the factor contrasts involved fidelity and cost issues, and in some cases a considerable cost difference was represented. Seastate/turbulence and pilot experience were added to enhance the generalizability of the experiment. Factors and levels are summarized in Table 1. The field-of-view "low" level was set to represent values for the SH-60B OFT (preliminary design values since the OFT was not yet operational), while the "low" setting for visual delay (217 msec) represents the slowest response time normally considered in simulators with visual systems. Both g-seat acceleration and vibration cueing were included in the experiment so that g-seat effects could be fully determined.

Performance Measurement

Raw data were recorded at 30 Hz and reduced to a set of trial summary measures. For measurement purposes, the task was partitioned into four segments. These segments were (1) the approach from 1500 feet astern to the stern of the ship, (2) transition from the stern to a hover above the landing point, (3) hover above

TABLE 1. EXPERIMENTAL FACTORS AND LEVELS

FACTOR	LEVELS "LOW"	"HIGH"
Scene Detail	Outline of deck & hangar	Full Deck & hangar markings, ship's wake and seascape patterns
Field of View	20 deg left to 100 deg right, 15 deg up to 25 deg down (left half field), and 3 deg up to 40 deg down (right half of field)	40 deg left to 120 deg right 20 deg up to 50 deg down
Visual Delay	217 msec	117 msec
G-seat Cueing	Off	Translation and angular accelerations
G-seat Vibration	Off	Oscillating cushions
Collective Sound	Off	Augmented aural cues
Seastate	Moderate seastate and medium air	Calm with no air turbulence
Average Flight Experience	830 hours	2323 hours

TABLE 2. SUMMARY OF EFFECTS

Factor	Effect Size	Segment/Measures	Best Option*
Scene Detail	Moderate/Large	All segments/most quality measures, pilot opinion	High detail
Visual Delay	Small/Moderate	Hover, touchdown roll, pitch control, pilot opinion	117 msec
Field of View	Small	Approach, hover touchdown/lineup control, aircraft pitch	Wide FOV
G-seat Vibration	Small	Approach, hover descent/stick lateral cyclic	?
G-seat Acceleration	None		?
Collective Sound	None		?

* The option that resulted in best simulator performance. In cases where quality measures were not affected, no determination of "best" performance was possible.

the landing point, and (4) descent to touchdown. Primary summary measures were root mean square (RMS) error from the prescribed flight path and touchdown error scores. Extensive pilot opinion data were also collected via questionnaires. More detail on performance measurement can be found in Westra and Lintern [11].

RESULTS I

Scene detail had by far the largest effect with most measures and all task segments affected. Performance was considerably better with the high detail ship and wake scene. Visual delay had the next largest effect of the equipment factors. There was a small but significant effect in favor of the shorter delay time. Pilot opinion strongly supported this effect. Field of view was ranked next in terms of overall effect magnitude with mostly small performance effects favoring the wide field of view. G-seat acceleration cueing, g-seat vibration cueing and collective sound had essentially no meaningful performance effects in the experiment. The effects are summarized in Table 2. Effect size refers to the degree of variability in performance which can be attributed to the factor listed.

DISCUSSION/RECOMMENDATIONS I

Ideally, if no particular problems with methods, measures, equipment and procedures were noted, the results of a performance experiment would feed directly into the planning and design for an in-simulator transfer experiment. Unfortunately, several problems came to light which suggested that further work was needed before moving to the in-simulator transfer experimental stage. First, despite extensive development work and pretesting, the g-seat acceleration cueing was judged by most pilots to be inaccurate and distracting. Also, performance differences were not noted with the g-seat cueing present. Therefore, more work was required to determine if g-seat cueing could affect performance.

Second, with the task performed continuously from start to touchdown, there were problems with widely varying amounts of time spent in the hover. Two pilots in particular typically came over the stern low and quickly landed, often resulting in little or no hover time during which data could be collected. Since the hover is considered a very important element of the task, this problem was considered serious enough to warrant a change in procedures so that hover data would be collected. Third, performance measurement was considered inadequate for fully documenting the visual delay and g-seat effects.

The scene detail effect indicated that the low detail scene tested was inadequate and it was recommended that this level not be studied further. It was also concluded that the longer visual delay was unacceptable for this task. Since performance effects were small, it was felt that the relatively narrow OFT field of view was adequate. However, it was noted that representation of the chin window area is important, and this area was not fully tested in the experiment. G-seat vibration and collective sound had no appreciable effects on performance but pilots seemed to like these features as indicated on the pilot opinion surveys. It was recommended that they be incorporated into the simulation and not studied further, since they are relatively inexpensive, add face validity to the simulation, and do not impair performance.

PERFORMANCE EXPERIMENT II

Based on the outcome of the first performance experiment, a second performance experiment for the helicopter shipboard landing task was planned and conducted. This research effort actually consisted of two separate experiments; an approach, hover, and landing task, and a precision hover task. The precision hover task was selected both to correct procedural problems by insuring a defined hover segment, and to enhance performance measurement. The hover task was set up to include wind gusts at specific times so that pilot reaction time and frequency response under different conditions could be measured. Other measures of aircraft control and activity were also added to the performance measure set to insure complete description of any effects. A complete description of this research effort and results is given in Westra, Sheppard, Jones, and Hettinger [9].

A number of developmental improvements were made to the g-seat cueing algorithms in an attempt to correct the deficiencies noted in the first experiment. In particular, the acceleration cueing drive algorithms were deemphasized with emphasis shifted to rate and position cueing. Further, the major cueing activity was focused in the vertical dimension. This strategy was derived from results given in McMillan, Martin, Flach, and Riccio [5]. G-seat vibration cueing was removed completely from the inflatable seat pads and presented via a mechanical seat shaker. In the first experiment, vibration cueing was presented through the seat pads along with acceleration cueing, and it was felt that this may have contributed to "overloading" the g-seat with more information than could reasonably be assimilated.

Tasks

Two tasks were defined and an experiment was performed for each task. The first task was defined as before with an approach, transition to hover, descent and landing. However, the hover portion of the task was altered to force a minimum of 20 seconds in a defined hover. Once pilots entered the defined hover segment, they were required to hold hover and not attempt a landing until given a green light. This procedure corrected problems with data collection in the hover segment during the first experiment. The second task was a 60-second precision hover over the landing deck during which three vertical wind gusts were presented, randomly either up or down.

Factors and Levels

Dynamic seat cueing was included as a factor for both tasks (on or off) while seat vibration cueing via the mechanical seat shaker was a constant condition. Visual delay was included as a factor for the precision hover task only at values of 183 and 117 msec. The 183 msec condition represents one frame less than the 217 msec investigated in the previous experiment. This is the next logical value to examine after

determining that 217 msec is unacceptable. A major software update to the aerodynamic model was also tested in this experiment and included as a factor for both tasks (updated model vs. standard model). This factor is referred to as dynamic inflow.

Field of view was included as a factor for both tasks. However, in these experiments, the low level values were based on actual measurements at the now operational SH-60B OFT, as opposed to preliminary design values used in the previous experiment. These values differed from the values used in Performance Experiment I in several respects, most critically in the downward field of view, which was approximately 9 degrees less. In addition, vertical dark areas present in the OFT but not modeled in the first experiment were included in the low level field of view. The high level field of view was the widest VTRS capability, 160 degree horizontal by 70 degree vertical.

Scene detail was included as a factor for both tasks, but in this experiment, the comparison was a VTRS model of the detail available in the SH-60B OFT versus a higher level of detail ship. The primary differences in detail were a VTRS higher detail wake, "pad eyes" on the deck of the high detail ship (an attempt to provide some texture for altitude cueing), plus added antennae and a ladder on the hangar wall of the high detail ship (see Figures 1 and 2). Seastate was used as a difficulty factor for the approach, hover and landing task and pilot experience was categorized as a factor in both tasks. A total of 12 experienced SH-60B pilots participated in the experiments. A summary of the factors and levels in the experiments is given in Table 3.

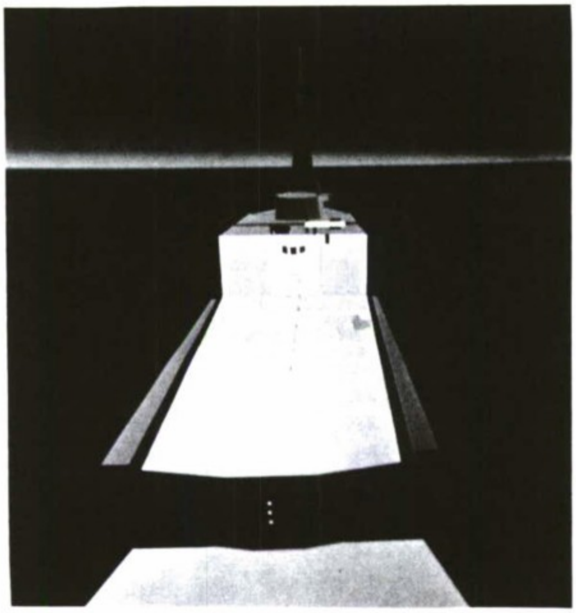


Figure 1. VTRS model of SH-60B OFT visual scene

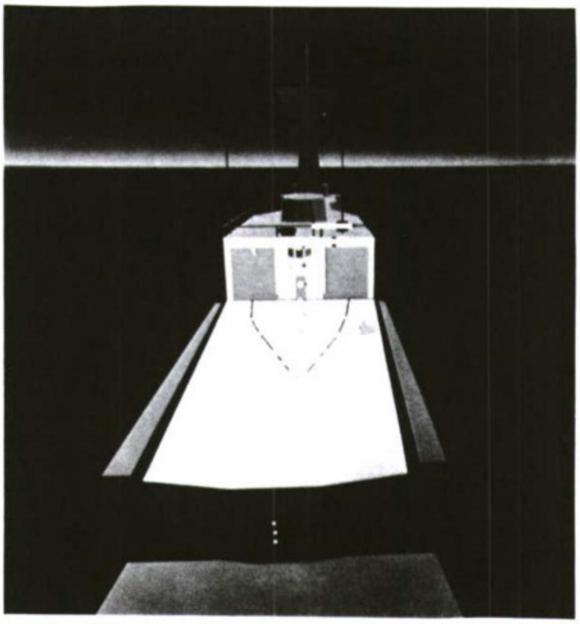


Figure 2. VTRS "high" detail scene.

RESULTS II

Results from the second research effort are summarized in Tables 4 and 5. By far the most striking result is the large field of view effect. In contrast to the fairly small effects that were found in the first experiment, the effects were pervasive (influencing all task segments and many measures) and strongly favored the VTRS wide field of view. Scene detail effects were more modest as expected with the only substantial effect (favoring the higher detail) occurring during the approach segment on lineup control.

The updated aerodynamic model proved beneficial as performance was enhanced for several measures. The visual delay factor had only a small effect on performance, affecting primarily roll activity in the hover, with greater roll variability evidenced with the longer delay. Dynamic seat cueing did not appear to have any meaningful performance effect. Pilot opinion was generally more favorable toward seat cueing than in the first experiment, but pilots still did not favor it over the no seat cueing condition.

DISCUSSION

The problems noted in the first performance experiment appear to have been resolved in the second performance experiment. In particular, problems with procedures for the hover segment were successfully resolved and performance

TABLE 3. SUMMARY OF EXPERIMENTAL FACTORS AND LEVELS

Approach, Hover, and Landing and Precision Hover Tasks

Factor	Level	
Scene Detail	Moderate detail (SH-60B OFT)	High detail (VTRS)
Field of View	Restricted (SH-60B OFT)	Wide (VTRS)
Dynamic Seat Cueing	G-seat Off	G-seat On
Dynamic Inflow	Standard Rotor Aerodynamic Model	Enhanced Rotor Aerodynamic Model

Approach, Hover, and Landing Task Only

Factor	Level	
Seastate	Moderate Seastate (2) and medium air turbulence	Calm Seastate and no air turbulence

Precision Hover Task Only

Factor	Level	
Visual Delay	183 msec	117 msec
Difficulty	Three distinct vertical gust disturbances (counterbalanced combination of up or down)	

TABLE 4. SUMMARY OF EFFECTS FOR THE APPROACH, HOVER, AND LANDING TASK

Factor	Effect Size	Segment/Measurement	Better Option*
Field of View	Large	Effects in all task segments across many measures	VTRS wide field of view
Dynamic Inflow	Moderate/ Small	Effects in glideslope during the approach and altitude control during hover	Enhanced Rotor Model
Scene Detail	Small	Effects in lineup and roll activity in the approach segment	Upgraded VTRS Scene
Dynamic seat Cueing	Small	Did not have a meaningful effect on performance	?
Seastate	Large	Difficulty factor performance was better without seastate	n/a
Pilot Differences	Large	Large control differences	n/a

*The option that resulted in better simulator performance. In cases where quality measures were not affected, no determination of "better" was possible.

TABLE 5. SUMMARY OF EFFECTS FOR THE PRECISION HOVER AND LANDING TASK

Factor	Effect Size	Segment/Measurement	Better Option*
Field of View	Large	Effects in all task segments across many measures	VTRS wide field of view
Scene Detail	Small	Effected pitch control in hover	?
Dynamic Inflow	Small	Response time to gusts	Enhanced model
Visual Delay	Small	Effected longitudinal and vertical positioning and roll activity in hover	117 msec
Dynamic seat Cueing	Small	No meaningful performance benefits with g-seat on	?
Pilot Differences	Large	Large control differences	n/a

*The option that resulted in better simulator performance. In cases where quality measures were not affected, no determination of "better" was possible.

measure inadequacies were also corrected. Procedurally then, and with regard to performance measurement, we are now ready to move into the in-simulator transfer-of-training research phase. G-seat cueing has been pursued through two extensive stages of development and refinement, and still appears to offer no real benefit to performance or even face validity for the shipboard landing task. It would seem that further research with seat cueing for this task is not likely to result in any meaningful pay off.

The large performance effects due to field of view were unexpected in light of the rather small effects observed in the first experiment. There are several probable sources contributing to this difference, the most likely being that the actual measured OFT field of view used in this research effort differed in some critical ways from the preliminary design values for the OFT field of view used in the first experiment. Most importantly, the downward field of view was approximately 9 degrees less for the OFT field of view in the second research effort. In addition, there were two vertical dark areas 5 degrees in width present for the OFT display in the second research effort but not the first. These dark areas are present in the OFT where there are spaces between display screens.

RECOMMENDATIONS

Based on the outcomes from the performance experiments a number of specific statements and recommendations can be made. The following recommendations have direct implications for the design of future transfer-of-training experiments at VTRS and design considerations for the SH-60B OFT:

1. The field of view in the OFT appears inadequate. The downward field of view in the chin window area is probably the most critical and any efforts to upgrade the OFT field of view should start in this area. However, since transfer of training is the ultimate issue in a training environment, not performance per se, final judgment should be withheld until the completion of a transfer-of-training experiment. Due to the large performance effects, and the high cost nature of the issue, it is recommended that field of view be included as a factor in the next in-simulator transfer-of-training experiment.

2. The scene detail in the OFT appears adequate with the exception of the wake. The first experiment gave results clearly indicating that a very low level of detail was not adequate for performing the task, so certainly less

detail than is present in the OFT cannot be recommended. Since this issue has been somewhat resolved on a performance level, there is not a clear need to carry this factor into the in-simulator transfer phase.

3. The aerodynamic model update should be incorporated at the OFT and at VTRS as a constant condition.

4. G-seat cueing should be dropped from further consideration for future VTRS research on helicopter shipboard landing.

5. G-seat vibration cueing should be incorporated as a constant condition at VTRS via the mechanical seat shaker.

6. A visual delay of 217 msec appears too long for this task. A delay of 183 msec is probably the longest delay that should be considered in a visual system for the task. Although performance effects compared to 117 msec are small, it is felt that 183 msec is only marginally acceptable for existing OFTs, and a shorter delay would be recommended for new acquisitions or upgrades of visual systems.

DEVELOPMENT FOR AN IN-SIMULATOR TRANSFER-OF-TRAINING EXPERIMENT

At the present time, planning and development for an in-simulator transfer-of-training experiment is underway at VTRS. The transfer-of-training experimental paradigm brings an additional dimension into focus, namely training. In this environment, not only equipment variables may influence the training process and subsequent transfer, but instructional variables also affect learning and transfer. In fact, previous experience at VTRS has suggested that instructional variables have potential for a greater impact on learning than equipment variables. Further, instructional variables may interact with equipment variables in such a way that equipment costs can be saved if certain instructional strategies are followed. For example, Westra [7] found that the carrier landing task can be more quickly trained under a backward chaining scheme than a whole task (from the abeam position) strategy. It was further determined that if the backward chaining scheme is used, a wide field of view is not necessary for the visual display.

For the experiment under development, the results of the previous performance research have been incorporated, and as a result of this, field of view will be included in the design. No other equipment factors will be directly manipulated since those issues were essentially resolved on a performance basis. However, several instructional variables are under consideration for inclusion in the experiment. These variables are number of training trials, task chaining, and augmented cueing. Task chaining involves segmenting the task and progressively adding segments until the whole task is presented. Thus, the hover and landing phase would be taught first, followed by the transition to hover, hover, and landing, and finally the whole task-approach transition to hover, hover, and landing. Augmented cueing refers to the use of artificial cues in addition to the normal cues already present.

SUMMARY

This paper has attempted to provide an overview of the research program on simulation and training for the helicopter shipboard landing task at VTRS. The overall research strategy incorporating a three phase process and holistic experimental philosophy was presented and discussed. Two major research efforts representing the first phase of research were presented and discussed. It was shown how the research proceeded in a logical manner with results used to build on previous findings and guide succeeding research. The implications of the first phase of research were discussed and it was shown how decisions were made to either make recommendations for simulator design or the next step of the research process. Finally, the development for the second phase of the research program, an in-simulator transfer-of-training experiment, was discussed. The issue of instructional strategies can be investigated in this stage of research, and the implications were discussed.

REFERENCES

1. Herndon, J.W. A Visual Technology Research Simulator for Vertical Take-off and Landing. Orlando, FL: Naval Training Systems Center, NAVTRASYSCEN IH-337, March 1982

2. Lintern, G., Sheppard, D. J., Mckenna, D., Thomley, K. E., Nolan, M., Wightman, D. C., and Chambers, W. S. Simulator design and instructional features for air-to-ground attack: Transfer study. Orlando, FL: Naval Training Systems Center, NAVTRASYSCEN 85-C-0044-1, 1985.

3. Lintern, G., Thomley, K., Nelson, B., and Roscoe, S. N. Content, variety, and augmentation of simulated visual scenes for teaching air-to-ground attack. Orlando, FL: Naval Training Systems Center, NAVTRASYSCEN 81-C-0105-3, 1984.

4. Lintern, G., Wightman, D. C., and Westra, D. P. An overview of the research program at the Visual Technology Research Simulator. Proceedings of the 1984 Image Conference III. Phoenix, AZ: The Air Force Human Resources Laboratory, June 1984.

5. McMillan, G. R., Martin, E. A., Flach, J. M., and Riccio, G. E. Advanced dynamic seats: An alternative to platform motion? Proceedings of the 7th Interservice/Industry Training Equipment Conference, Orlando, FL, November 1985.

6. Simon, C. W. New research paradigm for applied experimental psychology: A system approach. Westlake Village, CA: Canyon Research Group, Inc., Technical Report CWS-03-77, 1977.

7. Westra, D. P. Simulator design features for carrier landing. II. In-simulator transfer of training. Orlando, FL: Naval Training Systems Center, NAVTRASYSCEN 81-C-0105-1, December 1982.

8. Westra, D. P. Simulator design features for air-to-ground bombing: I. Performance experiment. Orlando, FL: Naval Training Systems Center, NAVTRASYSCEN 81-C-0105-4, September 1983.

9. Westra, D. P., Sheppard, D. J., Jones, S. A., and Hettinger, L. J. Simulator design features for helicopter shipboard landings. II: Performance experiments. Orlando, FL: Naval Training Systems Center, NAVTRASYSCEN 85-D-0026-9, July 1987.

10. Westra, D. P., Simon, C. W., Collyer, S. C., and Chambers, W. S. Simulator design features for carrier landing: I. Performance experiments. Orlando, FL: Naval Training Systems Center, NAVTRASYSCEN 78-C-0060-7, September 1982.

11. Westra, D. P., and Lintern, G. Simulator design features for helicopter landing on small ships. I. A performance study. Orlando, FL: Naval Training Systems Center, NAVTRASYSCEN 81-C-0105-13, 1985.

12. Westra, D. P., Lintern, G., Sheppard, D. J, Thomley, K. E., Mauk, R., Wightman, D. C., and Chambers W. S. Simulator design and instructional features for carrier landing: A field transfer study. Orlando, FL: Naval Training Systems Center, NAVTRASYSCEN 85-C-0044-2, June 1986.

ABOUT THE AUTHORS

DANIEL P. WESTRA earned his Ph.D. in Human Factors Psychology from the University of South Dakota in 1978. Since being employed at Essex Corporation, he has been at the Naval Training System Center's Visual Technology Research Simulator (VTRS). He is currently Program Manager for Essex Corporation's VTRS support research. His primary research interests are in the areas of economical multifactor experimental design and flight simulator design.

DENNIS C. WIGHTMAN is a research psychologist responsible for the Behavioral Research Program on the Visual Technology Research Simulator at the Naval Training Systems Center in Orlando, Florida. He received a Ph.D. in Psychology from the University of South Florida in 1983. His primary interests are in the areas of part-task training, perceptual motor skill, and the application of simulators to the training process.

JOYCE J. MADDEN has worked for the past two years as a research psychologist in the Human Factors Division of Naval Training Systems Center in the area of aviation simulation. Her most recent research has dealt with the SH-60B shipboard landing task. She also consults on various aricraft and ship acquisition programs at the Center. She has three years' experience as a human factors engineer at Naval Air Systems Command. Her Master's degree is in industrial psychology from George Mason University in Virginia.

EFFECT OF SCENE CONTENT AND FIELD OF VIEW ON WEAPONS DELIVERY TRAINING

Lt Kevin W. Dixon
Capt David G. Curry

Air Force Human Resources Laboratory
Williams AFB, Arizona

ABSTRACT

Two of the issues faced by designers of modern high-performance aircraft simulators are: (1) the level of visual scene realism required to adequately train complex tasks within the simulator; and (2) the field-of-view required for such training. The experiment discussed in this paper was designed to study both of these problems as they relate to the training of manual dive bombing in the F-16 aircraft. The experiment was performed in two separate simulators using the same visual image generators and data base. The first simulator was a Fiber Optic Helmet Mounted Display (FOHMD) System with a full 360 degree field of regard; the second used Wide Angle Collimated (WAC) Windows to provide a more restricted field-of-view (FOV). Subjects with no previous fighter aircraft experience were trained to perform 10°, 20°, and 30° dive bomb attacks on either a simulated bombing circle, a low detail airfield target scene, or a high detail simulation of the same scene. The transfer/test condition was a second different high detail airfield scene.

INTRODUCTION

The simulator can provide an ideal training ground for a wide variety of flight tasks. In years past, the tasks fell mostly in the realm of procedures training, cockpit layout familiarization, and basic contact/transition skills. As flight simulators become more complex and visual image generators more powerful, many new and previously unconsidered tasks can be trained in the simulator - complex skill and judgment tasks. For many such tasks the simulator may prove to be a better initial trainer than the actual aircraft, particularly those involving large amounts of initial approach and set-up time or high degrees of risk. An ideal candidate task for such training is precision ground attack - the actual aircraft can only carry a limited number of practice bombs and each pass requires considerable positioning time.

If such tasks are to be trained within the simulator, two questions are of prime importance: (1) how much detail must the visual scene contain; and (2) how large of a field-of-view does the pilot require to perform the task. More complex visual scenes require more powerful (and more expensive) image generators. Full FOV systems are much more complex than those with more limited fields of view (and approximately five times as expensive). Using the minimum effective capability is important for cost savings in both acquisition and maintenance.

BACKGROUND

Scene Content

Investigations of scene content variables and their effects on pilot performance are relatively few in number, and for the most part have concentrated on approach and landing and low altitude flight. Buckland, Monroe, and Mehrer (1980) placed a checkerboard textural pattern of various size directly on a runway.[2] Increased texture density produced greater control of the aircraft at touchdown, as indicated by slower vertical velocity, less displacement from centerline, and touchdown closer to the desired touchdown point. Kraft, Anderson, and Elsworth (1980) evaluated the effects of a complex visual scene, which included peripheral cues located adjacent to the runway.[7] The complex scene resulted in less vertical deviation from the glideslope for straight in approach segments, and less lateral deviation from centerline at touchdown. Additional research performed by Westra (1981, 1982) suggests that performance is enhanced in the approach and landing segments of flight when additional cues are presented.[13,14] It seems that further increases in scene complexity, particularly vertical object development along the approach and landing path, would result in further improvement to performance of this type of task.

Another task that has shown performance improvement with vertical development is low-level flight. Martin and Rinalducci (1983) used three terrain cue configurations--(1) all black inverted 35-foot high tetrahedrons, (2) inverted tetrahedrons of the same type with black bottoms and white tops, and (3) flat white triangles placed directly on the ground of the same density as conditions 1 and 3. The study showed that pilots performed better under conditions that had vertical development.[9] Another low-level flight study of interest was performed by Buckland (1981) to examine the effect of vertical cues and checkerboard textural patterns on flight performance. The results showed less deviation from the ideal flight parameters for the conditions involving vertical objects or textural cues.[1]

Some other interesting investigations into the effect of using the scene content variable in simulation studied its effect on performance for carrier landings and 30° bombing attacks.[15,8] Westra manipulated the scene content on a simulated carrier representation by using a day carrier scene as a high detail condition and a night carrier scene for a low detail condition. He found no transfer advantage between those trained with the high detail scene or the low detail scene. These findings suggested that a low detail scene could be used to train naval pilots for carrier landings. Lintern employed the same approach to study 30 degree bombing attacks. He used a complex day scene, bombing range with vertical objects, for the high detail condition and a dusk scene, bombing range less many terrain features, for the low detail condition. The experiment found no significant difference in either condition, but methodological problems between the comparisons

were discussed that could have confounded the data.

The experiment discussed in this paper was concerned primarily with the addition of two and three dimensional cues of known size to an otherwise simple data base. All three of the training data bases involved utilized irregular hardware texture to represent fields and other ground cues. The low and high detailed training data bases included two dimensional objects representing airport runways and three dimensional cues representing associated structures. The high detail testing data base used essentially the same representation for two and three dimensional cues as the high detail training data base (See Figures 1-4).

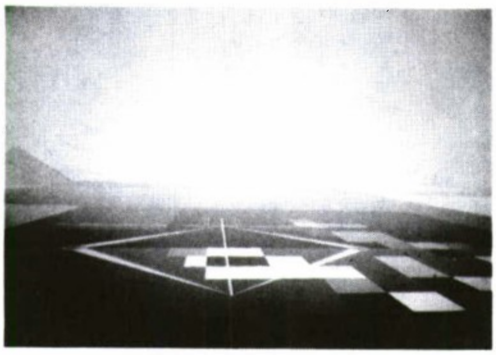


Figure 1. Standard Air Force Gunnery Range

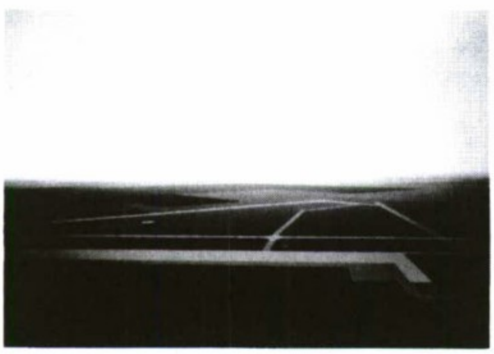


Figure 2. Low Detail Representation of Standard Air Force Runway (Cannon AFB NM)

Figure 3. High Detail Representation of Standard Air Force Runway (Cannon AFB NM)

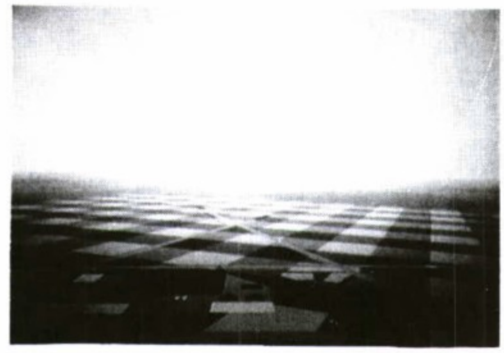
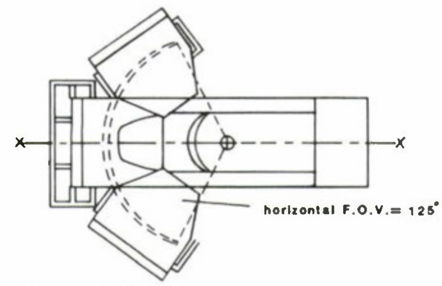


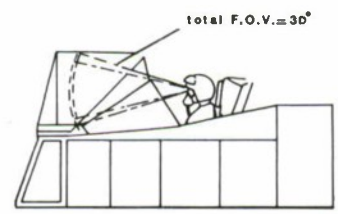
Figure 4. Test Condition: High Detail Representation of Standard Air Force Runway (China Lake NAS, CA)

Field-of-View.

The definition of FOV for this research effort will be the instantaneous field displayed by the system from the pilots eyepoint. For the current experiment and all subsequent mentioning of FOV dimensions will be in degrees with the pilots' eyepoint considered 0, 0 (See Figures 5-6).



(A) Plan view of display heads showing F.O.V. .



(B) Cross section through x-x showing vertical F.O.V. .

Figure 5: WAC window field of view size.

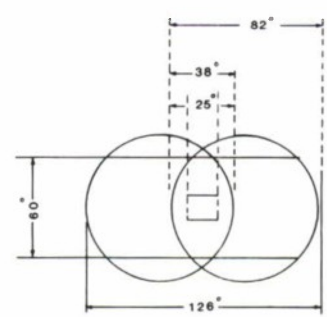


Figure 6: FOHMD Instaneous field of view size.

Many researchers have attempted to define FOV requirements. These early attempts focused almost exclusively on using the aircraft as the primary tool to provide data for FOV requirements.[18,12] These techniques incorporated either pilot subjective data or video techniques, such as mounting a camera in the cockpit that followed the pilot's eye-track. More recently attempts have focused on using the simulator as the primary research tool.

The early investigations utilizing the simulator as a research tool concentrated on determining the FOV requirements for straight-in take-offs and landings using experienced pilots. The results of these findings are summarized by Collyer, Ricard, Anderson, Westra, and Perry (1980)[3] as follows: safe and acceptable take-offs and landings could be performed in FOV configurations with dimensions of 10° horizontal by 10° vertical, 21.5° horizontal by 21.5° vertical, and 5.7° horizontal by 37° vertical. The most important findings of this series of studies were that the FOV configurations used were significantly smaller than those currently used in simulation. Other studies on take-offs and landings comparing performance across two FOV's concluded that no significant differences were noted between a 36° horizontal by 48° vertical limited FOV and a 300 horizontal by 150° vertical FOV for take-offs using experienced pilots.[6] These results are consistent.

These early investigations led to research designed to investigate basic contact maneuvers. This strategy was investigated to provide further options for the training of Air Force pilots, since fuel and aircraft costs had risen considerably. Several studies were accomplished that explored basic contact maneuvers performed by undergraduate pilot students, such as aileron rolls, barrel rolls, and the 360° overhead (OVHD) landing pattern. In three subsequent studies, between 1977 and 1979, FOV was used as a independent variable in conjunction with various other environmental factors.[6,5,11] These studies showed that the FOV requirements are extremely maneuver specific, but performance improved as the FOV increased. Several of the other variables investigated in these studies were significant across the various tasks and several did not affect performance in any manner.

The significant technological advances of the last decade in simulator design and display mediums began driving FOV research toward defining more complex tasks that could now be accomplished in the simulator (air-to-ground attacks, aerial refueling, carrier landings, and close air support). These studies examined the effect of various FOV configurations on experienced pilots for each of the above specified maneuvers.[4,8,17] All of the above studies showed that a FOV smaller than those in the actual aircraft could be used to practice these tasks by experienced pilots. A general summary of the significant results was stated by Wiekhorst and Vaccaro (1986) as follows: (a) flying tasks can be performed with a LFOV or area of interest in the simulator, and (b) the FOV requirement is very task.[16] Although consistent research has been completed on the FOV requirement, there needs to be significant in-simulator transfer of training studies using inexperienced pilots with larger sample sizes then previous studies. The current experiment will investigate the effects of FOV on acquisition and in-simulator transfer.

OBJECTIVE

The objective of this research was to determine the effect of scene content and field of view size on the skill acquisition of manual dive bombing tasks.

Subjects

Thirty-six current Air Force pilots with high performance, Fighter/Attack/Reconnaissance (FAR) aircraft ratings were used as subjects in this experiment. None of the subjects had any previous flight experience in the F-16 aircraft or previous dive bombing experience. All were currently flying the Northrop T-38 aircraft, a supersonic flight trainer.

Apparatus

This study was conducted in an F-16C flight simulator with two visual display systems. The first was a window type display using three wide angle Collimated (WAC) windows with an approximate field of view of 125° horizontally and 36° vertically. The second display was a Fiber Optic Helmet Mounted Display (FOHMD) which allowed an unrestricted field of view in all directions (cockpit, wings, nose were computer masked), and an instantaneous FOV of 126° horizontally and 60° vertically, with the only restricted visual area being that occupied by the simulated aircraft itself. Imagery in both cases was provided by a Singer-Link Digital Image Generation System (DIGS). Identical data bases were used under both of the display conditions. The cockpit itself was fully instrumented, with the Head-Up Display (HUD) targeting system set for the manual bombing mode. All necessary information to perform the dive bombing tasks (dive angle, airspeed, g-factor, altitude, flight path marker, targeting reticle, and compass heading) was presented to the subject in the HUD, thus lessening the distraction caused by having to perform complex cross checks with an unfamiliar cockpit layout.

Experimental Design

This study was a mixed design. Independent variables and their treatments were as follows:

1. Field-of-View:

   (a) WAC Window (125° x 36°)
   (b) FOHMD (360° field-of-regard)

2. Scene Content (Data Base):

   (a) Low Detail Bombing Range
   (b) Low Detail airfield
   (c) High Detail airfield

3. Presentation Order:
   (a) 10°, 20°, 30°
   (b) 20°, 30°, 10°
   (c) 30°, 20°, 10°
   (d) 30°, 10°, 20°
   (e) 20°, 10°, 30°
   (f) 10°, 30°, 20°

4. Dive Angle:
   (a) 10°
   (b) 20°
   (c) 30°

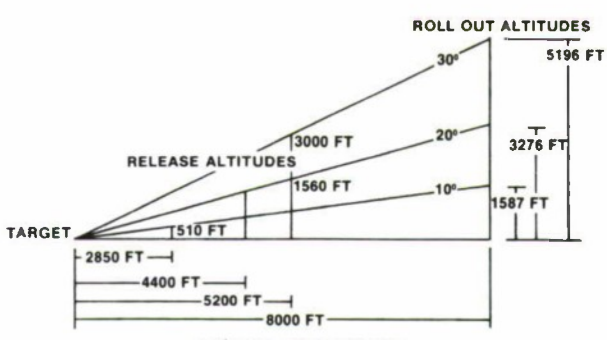
The training data bases on this study varied on the amount of visual information they presented to the pilot. The lowest level was a standard Air

Force gunnery range with minimal visual information, primarily a target circle with three down range distance markers at 600, 1250, and 2000 feet. The second data base was a two dimensional representation of Cannon AFB, NM including all runways, taxiways, parking aprons, etc. The third data base differed from the second only by the addition of three dimensional cues around the airfield (buildings, hangers, a control tower, etc). The target in both of the airfield conditions was located at the intersection of a taxiway and the main runway. (Refer to Figures 1-4.) Each subject performed attacks at dive angles of 10°, 20°, and 30°. The order of presentation of these angles was balanced across subjects, with each subject completing all passes at a given dive angle before proceeding to the next. Dive angles were investigated within subjects. Data bases, field-of-view, and dive angle presentation order were looked at across subjects.

Procedure

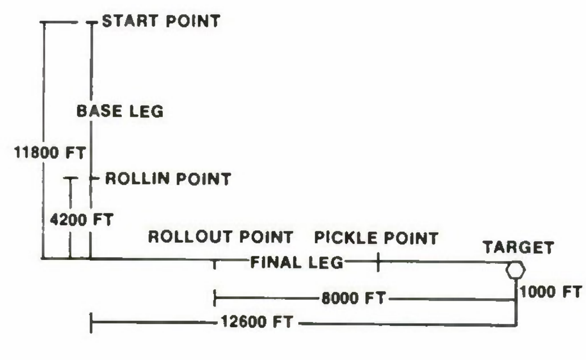
Each subject was given a one-half hour briefing on the nature of the experiment and on the techniques for performing a manual dive bomb attack in the F-16 aircraft. This briefing included information on both optimum delivery patterns and parameters, as well as correction factors for variations from optimum. During this briefing, the subject was also familiarized with the Heads-Up Display and instrumentation within the simulator cockpit. Following this, the subjects were put into the simulator and instructed to practice the material which they had just learned. A practice pass consisted of the simulated aircraft being placed on base leg of the bombing run 11,800 feet back from the target and 12,000 feet outboard from it, initialized at an altitude commensurate with the dive angle to be used in the attack (2500', 4000' or 7000'). Ideally, the subject was to maintain straight and level flight until reaching a point parallel to the target, then make a 90° turn toward the target, rolling out of the turn on the prescribed dive angle. During the dive, he was to accelerate to 450 knots, align the target reticle with the target, and release the bomb at the proper altitude (500', 1600' or 3000' depending on dive angle).

FIGURE 7: FINAL LEG PARAMETERS FOR DIVE BOMB TASKS



*ALL ALTITUDES ARE ABOVE GROUND LEVEL (AGL)
NOT TO SCALE

FIGURE 8: IDEAL FLIGHTPATH



NOT TO SCALE

Following each training passes, feedback was provided by the experimenter on how well the subject followed the commanded flight profile, bomb miss distance, the angle at which the bomb landed with respect to the target, deviation from ideal release parameters (dive angle, airspeed, and release altitude), and possible corrective actions which might have been implemented. Subjects continued flying training passes until reaching a predetermined criterion level of performance defined as three successive training passes resulting in bombs falling within 65 meters of the target. Once this level of proficiency was attained, the subject was transitioned to a test condition which consisted of a series of six simulated attacks on a second high-detail airfield (China Lake NAS, CA). In this condition subjects were provided with feedback only on their miss distance and miss angle. Each dive angle was both trained and tested before proceeding on to the next dive angle.

Data Analysis

Data from this research effort were initially examined using the SPSSX-MANOVA program resident on the AFHRL/OT VAX 11/780 computer system. In cases where the MANOVA indicated significant results, further examination of the univariate ANOVAs were performed and residuals were looked at using the RUMMAGE statistical package. With the MANOVA procedure, the Wilks F-statistic was used to determine whether a multivariate effect had reached significance. Only comparisons of a priori importance were investigated. These were: (1) comparison of both of the data bases exhibiting no vertical development (the bombing range and the low detail Cannon AFB data base); (2) comparison of the two Cannon data bases to determine the effect of adding the three dimensional cues; and (3) comparison of the bombing range and the high detail Cannon data base.

Data from the experiment were divided into three categories. The first set included the number of practice trails needed by the subject to reach the required minimum level of proficiency. The second set included data relating to the subject's approach profile during the series of test attacks. The final set was composed of the instantaneous parameters from the aircraft at the moment the bomb was released.

A. Training trials:

(1) Total number of training passes across all dive angles.

(2) Number of training passes required for 10° proficiency.

(3) Number of training passes required for 20° proficiency.

(4) Number of training passes required for 30° proficiency.

B. Approach to the target:

(1) Mean and standard deviation of roll.

(2) Mean and standard deviation of pitch error.

(3) Mean and standard deviation of g's.

(4) Mean and standard deviation of the horizontal deviation.

(5) Mean and standard deviation of the altitude deviation.

(6) Mean and standard deviation of airspeed.

C. Bomb release parameters:

(1) Roll.

(2) Pitch error.

(3) g factor.

(4) Horizontal deviation from ideal flight path.

(5) Deviation from ideal bomb release altitude.

(6) Airspeed.

(7) Bomb miss distance from the target.

RESULTS

<u>Comparison I: Bombing Range vs Low Detail Cannon</u>

A. Trials Data: No significant effects were noted for any of the training trials metrics.

B. Approach Data: The FOV by dive angle and the data base by dive angle interactions were both significant ($F(24,58)=1.693$, $p=.050$, and $F(24,58)=1.889$, $p=.025$), as was the dive angle effect ($F(24,58)=32.804$ $p<.0005$). For the FOV by dive angle interaction, the univariate F tests showed the effect was in the mean altitude deviation and standard deviation of roll metrics ($F(2, 40)=3.866$, $p=.029$ and $F(2, 40)=3.415$, $p=.049$). Graphic representation of this data is presented in Figures 9 and 10. The data base by dive angle effect lay solely in the mean altitude deviation metric ($F(2,40)=4.354$, $p=.019$). This effect is shown in Figure 11. All metrics for the dive angle effect were significant with the exception of mean airspeed and the standard deviation of the horizontal flight path deviation (see Table 1).

Figure 9: Field of View by Dive Angle Interaction
Mean Altitude Deviation

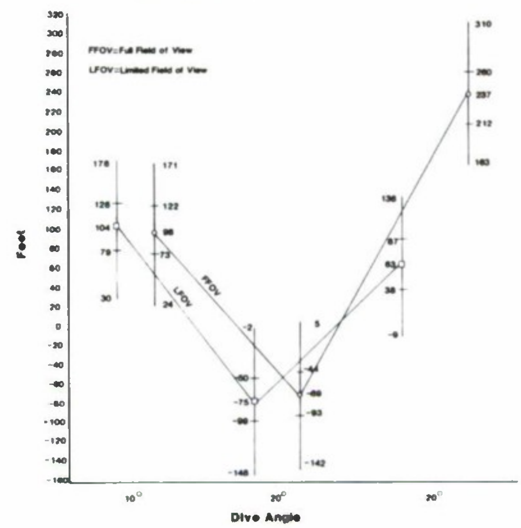


Figure 10: Field of View by Dive Angle Interaction
Standard Deviation of Roll

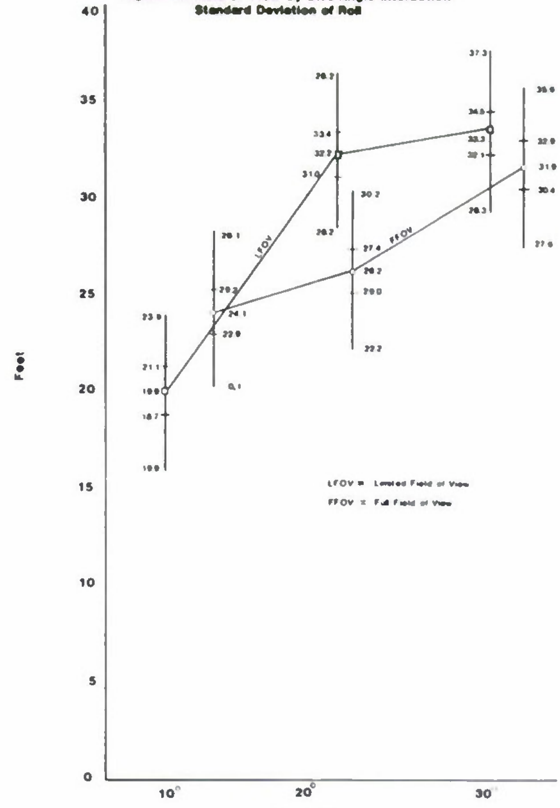
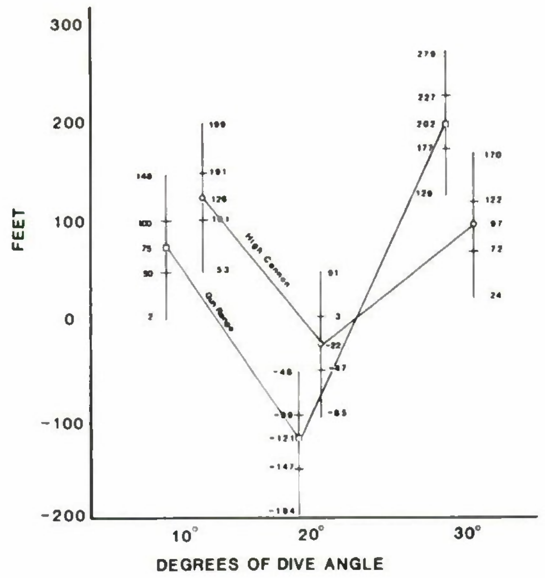


Figure 11: Data base by Dive Angle Interaction, Mean Altitude Deviation



	10°	20°	30°
MEAN ROLL	-7.3 deg	-6.6 deg	-12.0 deg
MEAN PITCH ROLL	.9 deg	6.5 deg	7.6 deg
MEAN G	1.30 G	1.69 G	1.99 G
MEAN HORIZONTAL FLIGHT PATH ERROR	428 ft	361 ft	812 ft
MEAN GLIDE SLOPE ERROR	101 ft	-71 ft	150 ft
STD DEV ROLL	22.0 deg	29.2 deg	32.4
STD DEV PITCH	3.2 deg	8.0 deg	11.4 deg
STD DEV G	1.10 G	1.81 G	2.30
STD DEV ALTITUDE ERROR	128 ft	234 ft	364 ft
STD DEV AIRSPEED	17 kt	17 kt	24 kt

Table 1: Effect of Dive Angle on Approach

C. Release Data: Analysis of the release data showed significant effects for FOV (F7,14)=3.265, p=.028), data base (F(7,14)=4.904, p=.006), and dive angle (F(14,68)=10.270, p<.0005). For the FOV effect, the univariate F tests showed the effect was in the roll metric and the horizontal flight path deviation metric (F(1,20)=6.382, p=.020 and (F1,20)=9.649, p=.006). Pilots with limited fields of view exhibited an average of 3° of right roll and were almost 420 feet to the left of the ideal ground track at bomb release, while those with full fields of view averaged 2° of left roll and were only 110 feet off track, also to the left. This is summarized in Table 2. The significant metrics for the data base effects were roll (F(1,20)=7.249, p=.014), pitch (F(1,20)=7.753, p=.011), and horizontal deviation (F1,20)=6.536, p=.019. Pilots trained on the bombing range averaged of 2° of left roll, were pitched 1.7° shallower than optimum, and were 144 feet off to the left at bomb release. Pilots trained on the low detail Cannon AFB data base averaged 3° of right roll, .35° of pitch error, and 390 feet of ground tract deviation at this point (see Table 3). For the dive angle effect, all metrics other than roll were significant. Results for this effect are summarized in Table 4.

	LIMITED FOV	FULL FOV
ROLL	-3.1 deg	1.9 deg
HORIZONTAL FLIGHT PATH ERROR	419 ft	109 ft

Table 2: Effect of Field of View on Release parameters

	BOMBING RANGE	LOW DETAIL CANNON
ROLL	2.0 deg	-3.3 deg
PITCH ERROR	1.7 deg	.3 deg
HORIZONTAL FLIGHT PATH ERROR	136 ft	392 ft

Table 3: Effect of Database on Release Parameters

	10°	20°	30°
SPEED	450 kts	456 kts	471 kts
PITCH ERROR	-1.0 deg	2.5 deg	1.5 deg
G	.86 G	1.18 G	1.10 G
HORIZONTAL FLIGHT PATH ERROR	-180 ft	220 ft	405 ft
RELEASE ALTITUDE ERROR	-4 ft	224 ft	321 ft
MISS DISTANCE	46 m	65 m	66 m

Table 4: Effect of Dive Angle on Release Parameters

Comparison II: Low Detail vs High Detail Cannon AFB

A. Trials Data: No significant effects were noted for any of the training trials metrics.

B. Approach Data: Significant effects were found for FOV ($F(12,9)=9.950$, $p=.001$), data base ($F(12,9)=3.388$, $p=.038$), and dive angle ($F24,60)=18.755$, $p<.0005$). For the FOV effect, the univariate F tests showed the effect was concentrated in the mean roll and mean horizontal deviation from the flight path ($F(1,20)=12.717$, $p=.002$ and $F(1,20)=32.722$, $p<.0005$). For the roll metric, pilots with a limited FOV averaged 1.5^{o} of right roll, while those with full FOV averaged only 6^{o}. For the horizontal deviation metric, limited fields produced an average of 925 feet of error off the ideal path, while full fields resulted in just over 40 feet of deviation (see Table 5). For the data base effect, significance was found on the G metric ($F(1,20)=4.670$), $p=.043$), mean horizontal path deviation ($F(1,20)=10.789$, $p=.038$). For the G metric, it was found that high detail trainees averaged 1.81G, while those trained with low detail averaged 1.55G. On the flight path metric, those trained on the higher detail condition were an average of 230 feet off track, while those trained on low detail were almost 740 feet off. Standard deviation of airspeed for high detail pilots was 22.5 knots, and 18.5 knots for low detail pilots. This is summarized in Table 6. All metrics for the dive angle effect were significant with the exception of mean roll, mean airspeed, and horizontal flight path deviation (See Table 7). Performance decreased as five angle increased.

	LIMITED FOV	FULL FOV
MEAN ROLL	-15.5 deg	-6.0 deg
MEAN HORIZONTAL FLIGHT PATH ERROR	925 ft	41 ft

Table 5: Effect of Field of View on Release Peremeters

	LOW DETAIL CANNON	HIGH DETAIL CANNON
G FACTOR	1.55 G	1.81 G
STD DEV AIRSPEED	19 kt	23 kt
MEAN HORIZONTAL FLIGHT PATH ERROR	737 ft	230 ft

Table 6: Effect of Database on Release Parameters

	10°	20°	30°
MEAN PITCH ERROR	1.0 deg	6.5 deg	7.3 deg
MEAN G	1.36 G	1.72 G	1.96 G
MEAN HORIZONTAL FLIGHT PATH ERROR	472 ft	326 ft	652 ft
MEAN GLIDE SLOPE ERROR	89 ft	-61 ft	107 ft
STD DEV ROLL	23.2 deg	30.6 deg	35.0 deg
STD DEV PITCH	3.6 deg	8.0 deg	11.0 deg
STD DEV G	1.14 G	1.87 G	2.31 G
STD DEV ALTITUDE ERROR	153 ft	233 ft	385 ft
STD DEV AIRSPEED	16 kt	20 kt	24 kt

Table 7: Effect of Dive Angle on Approach

C. Release Data: Examination of the instantaneous release point data, showed significant effects for FOV ($F(7,14)=5.156$, $p=.004$), data base ($F(7,14)=2.976$, $p=.039$), and dive angle ($F(14,68)=10.596$, $p<.0005$). For the FOV effect, significance was found for the horizontal flight path deviation metric only ($F(1,20)=27.266$, $p<.0005$), with full fields producing 45 feet of error at release versus 490 feet for the limited field. For the data base effect, the airspeed and horizontal deviation metrics are significant ($F(1,20)=5.035$, $p=.036$ and $F(1,20)=8.722$, $p=.008$). The low detail pilots averaged being 8 knots fast at release and were about 393 feet wide to the left of optimum. High detail pilots were approximately 17 knots fast, but only 140 feet wide. All of the dive angle metrics except roll were significant (See Table 8).

	10°	20°	30°
SPEED	451 kts	463 kts	473 kts
PITCH ERROR	0.7 deg	2.4 deg	1.4 deg
G	.91 G	1.21 G	1.11 G
HORIZONTAL FLIGHT PATH ERROR	-176 ft	-205 ft	-414 ft
RELEASE ALTITUDE ERROR	13 ft	261 ft	491 ft
MISS DISTANCE	50 m	64 m	70 m

Table 6: Effect of Dive Angle on Release Parameters

Comparison III: Bombing Range vs High Detail Cannon AFB

A. Trials Data: Again, none of the training trials data showed significant results.

B. Approach Data: The only significant treatment effect found was for dive angle ($F(24,60)=21.498$, $p=.0005$), with all metrics other than mean roll, mean airspeed, and the standard deviation of horizontal flight path deviation reaching significance (See Table 9). Results in general followed the previously observed pattern of better performance at shallower dive angles.

	10°	20°	30°
MEAN PITCH ERROR	1.3 deg	7.0 deg	7.7 deg
MEAN G	1.43 G	1.87 G	2.06 G
MEAN HORIZONTAL FLIGHT PAT ERROR	219 ft	90 ft	329 ft
MEAN GLIDE SLOPE ERROR	64 ft	-110 ft	160 ft
STD DEV ROLL	22.9 deg	31.3 deg	33.3 deg
STD DEV PITCH	3.9 deg	8.7 deg	12.0 deg
STD DEV G	1.21 G	1.84 G	2.35 G
STD DEV ALTITUDE ERROR	137 ft	256 ft	373 ft
STD DEV AIRSPEED	18 kt	21 kt	25 kt

Table 9: Effect of Dive Angle on Approach

C. Release Data: The FOV by data base interaction and dive angle effects were both found to be significant in this condition ($F(7,14)=3.167$, $p=.031$ and $F(14,68)=7.506$, $p<.0005$). For the FOV by data base interaction, significance was

concentrated in the roll and miss distance metrics ($F(1,20)-15.898$, $p=.001$ and $F(1,20)=5.365$, $p=.031$). These interactions are shown in Figures 12 and 13. For the dive angle effect, all of the metrics other than roll and miss distance were significant. These results are summarized in Table 10.

Figure 12: Field of View by Data base Interaction, Roll Metric

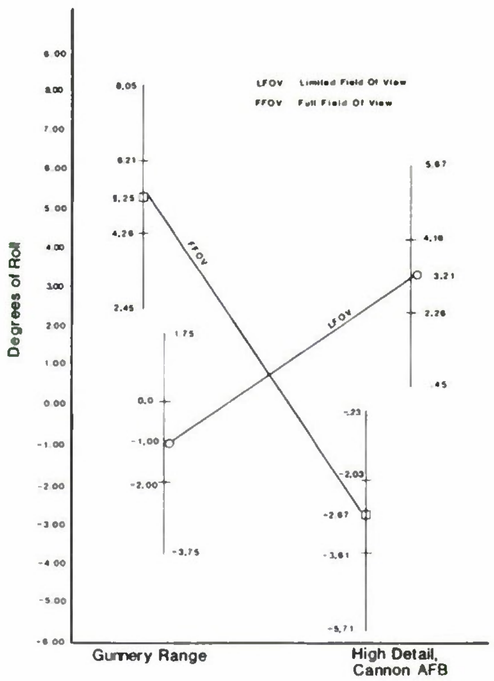
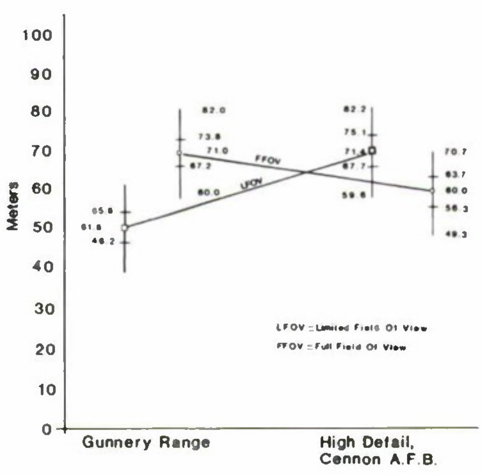


Figure 13 : Field of View by Data base Interaction, Bomb Miss Distance



	10°	20°	30°
SPEED	453 kts	463 kts	474 kts
PITCH ERROR	0.0 deg	3.4 deg	1.7 deg
G	.85 G	1.18 G	1.10 G
HORIZONTAL FLIGHT PATH ERROR	107 ft	62 ft	246 ft
RELEASE ALTITUDE ERROR	-1 ft	287 ft	395 ft
MISS DISTANCE	54 m	85 m	71 m

Table 10: Effect of Dive Angle on Release Parameters

DISCUSSION

This study was conducted to determine the effect of scene content and field-of-view on weapons delivery training. Neither scene content or FOV variables had a significant impact on the number of trials to reach proficiency. The approach data results revealed significant effects and interactions on a number of variables. There was a significant main effect associated with the task factor (10°, 20°, and 30° dive angle tasks) for approach and release data. In general performance was better with shallower dive angles. This can be attributed to the fact that the steeper dive angles are generally considered more difficult because release distances and altitudes are displaced further from the target.

Other significant main effects were noted for FOV and scene content in the approach and release data for - (1) the high versus low detail Cannon AFB comparison and (2) the release data in the bombing range versus low detail Cannon AFB. For FOV, this was reflected in a 10% larger horizontal deviation in the limited FOV condition. We believe that this is due to the difficulty associated with finding the proper roll-out cues, which are not visible at the turn point in the limited FOV condition. For scene content, the high detail airfield (with vertical development) was associated with a 70% decrease in horizontal deviations. The presence of buildings provided more precise cues for judging roll-out and run-in lines. The high detail airfield was also associated with higher g's at pull-out. We believe this effect may be due to the pilots' ability to better detect ground proximity with the addition of vertical development.

The main interaction effects for the approach data were FOV by dive angle and data base by dive angle in the bombing range versus low detail Cannon AFB comparison. The FOV by dive angle effect was concentrated in altitude deviation and roll. These effects are not easily interpreted. This effect did not appear in any of other comparisons and this is the only comparison involving scenes with no vertical development. It is possible that some other difference between the scenes manifest themselves in the absence of vertical cues. The data base by dive angle interaction was due to differences in altitude deviation, but the pattern, although consistent, allows no readily interpretable explanations (refer to Figures 9, 10, and 11).

The other main interaction effect was FOV by dive angle in the release data for the bombing

range versus high detail Cannon AFB comparison. The effect was due to the roll and bomb miss distance variables. Full FOV was associated with significantly more roll deviation for the gunnery range and less for the high detail airfield. We believe this is caused by pilots maneuvering more in an attempt to locate cues in the bombing range. The presence of vertical development in the high detail scene gave the pilot the appropriate cues and degrees of roll deviations decreased.

Even though there were no strong and consistent effects in bomb scores, the overall performance of subjects was better in training conditions that incorporated familiar objects (taxiways, aprons, and runway width) and vertical development. This is seen in the better adherence to the desired flight profile in the test condition for those trained in those conditions. There was also better performance for the full FOV display. This leads us to believe that tasks requiring close adherence to a command flight profile should use full FOV displays and incorporate vertically developed cues. Further testing is planned to validate this recommendation on additional air-to-ground maneuvers for scene content. Other follow-on investigations will include training in air-to-air and formation flight in limited FOV displays and testing in full FOV displays. This will help determine the transfer of training between the various FOV configurations.

REFERENCES

[1]Buckland, G.H., Edwards, B.J., and Stephens, S. Flight Simulator Visual Development for Terrain Flight Simulation. In Proceedings of the 1981 Image Generation/Display Conference II, June 1981, 351-362. AFHRL-TR-81-48, AD-A110 226. Williams AFB, AZ: Operations Training Division, Air Force Human Resources Laboratory.

[2]Buckland, G.H., Monroe, E.G., and Mehrer, K.I. (Aug 1980). Flight Simulator runway visual textural cues for landing (AFHRL-TR-79-81, AD A089 434). Williams AFB, AZ: Flying Training Division, Air Force Human Resources Laboratory.

[3]Collyer, S.C., Ricard, G.L., Anderson, M., Westra, D.P., and Perry, R.A. (Jun 1980). Field of view requirements for carrier landing training (AFHRL-TR-80-10, AD-A087 012). Williams AFB, AZ: Operations Training Division, Air Force Human Resources Laboratory.

[4]Hughes, R.G., and Brown, L. (May 1985). Flight simulator: effects of visual display field of view on A-10 aircraft close air support performance (AFHRL-TR-84-58). Williams AFB, AZ: Flying Training Division, Air Force Human Resources Laboratory.

[5]Irish, P.A., III, and Buckland, G.H. (June 1978). Effects of platform motion, visual and G-seat factors upon experienced pilot performance in the simulator (AFHRL-TR-78-9, AD A-055 691). Williams AFB, AZ: Flying Training Division, Air Force Human Resources Laboratory.

[6]Irish, P.A.,III, Grunzke, P.M., Gray, T.H. and Waters, B.K. (April 1977). The effects of system and environmental factors upon experienced pilot performance in the advanced simulator for pilot training (AFHRL-TR-77-13, AD A043 195). Williams AFB, AZ: Flying Training Division, Air Force Human Resources Laboratory.

[7]Kraft, C.L., Anderson, C.D., and Elsworth, C.L. (May 1980). Psychophysical criteria for visual simulation systems (AFHRL-TR-79-30, AD A084 776). Williams AFB, AZ: Flying Training Division, Air Force Human Resources Laboratory.

[8]LeMaster, D.W., and Longridge, T.M. (May 1978). Area of interest/field of view research using ASPT (AFHRL-TR-78-11, AD-A055 692). Williams AFB, AZ: Flying Training Division, Air Force Human Resources Laboratory.

[9]Martin, E.L., and Rinalducci, E.J. Low Level Flight Simulation: Visual Cues, Sep 83. AFHRL-TR-83-17, Williams AFB, AZ: Operations Training Division, Air Force Human Resources Laboratory.

[10]Nataupsky, M., Waag, W.L., Weyer, D.C., McFadden, R.W., and McDowell, E. (Nov 1979). Platform motion contributions to simulator training effectiveness: Study III-interaction of motion with field of view (AFHRL-TR-79-25, AD-A078 426). Williams AFB, AZ: Flying Training Division, Air Force Human Resources Laboratory.

[11]Weaver, A.J., Loikith, G.J., and Jordan, R.J. (Dec 1978). Design, development, and validation of field of view evaluation (NATC-SY-126R-78, AD-B033 411L). Naval Air Test Center, Patuxent River, MD.

[12]Weaver, A. J., Loikith, G. J., and Jordan, R. J. (Dec 1978), Design, Development, and validation of field of view evaluation (NATC-SY-126R-78, AD-B033 411L). Naval Air Test Center, Patuxent River, MD.

[13]Westra, D.P., Simon, C.W., Collyer, S.D., and Chambers, W.S. Simulator design features for carrier landing: I. Performance experiment. Orlando, FL: Naval Training Equipment Center, NAVTRAEQUIPCEN 78-C-0060-7, December, 1981.

[14]Westra, D. P. Simon, C.W., Collyer, S.C., and Chambers, W. S. Simulator design features for carrier landing: II. In-simulator transfer of training. Naval Training Equipment Center, NAVTRAEQUIPCEN 81-C-0105-1, December 1982

[15]Westra, D.P. Lintern, G., Sheppard, D. J., Thomley, K.E., Mauk, R., Wightman, D.C., and Chambers, W.S. Simulator design and instructional features for Carrier Landing: A Field Transfer Study. Naval Training Systems Center, NAVTRASYSCEN 85-C-0044-2, 1985

[16]Wiekhorst, L.A., and Vaccaro, F.T. (Sep 1986). Flight Simulator Field of view utilized in performing tactical manuevers (AFHRL-TP-86-29). Williams AFB, AZ: Flying Training Division, Air Force Human Resources Laboratory.

[17]Woodruff, R.R., Longridge, T.M., Jr., Irish, P.A., III, and Jeffreys, R.T. (May 1979). Pilot performance in simulated aerial refueling as a function of tanker model complexity and visual display field-of-view (AFHRL-TR-78-98, AD-A070 231). Williams AFB, AZ: Flying Training Division, Air Force Human Resources Laboratory.

[18]Yeend, R., and Carico, D. (Sep 1978). A program for determining flight simulator field-of-view requirements (NATC-TM 78-1-RW,ADA A058 932). Naval Air Test Center, Patuxent River, MD.

ABOUT THE AUTHORS

Kevin W. Dixon received his commission in the Air Force in 1985 after earning a BS in Experimental Psychology/Human Factors from the Air Force Academy. He is a Research Psychologist at AFHRL/OTE and is currently working on training effectiveness requirements for flight simulators.

David G. Curry is a Research Psychologist at AFHRL/OTE whose work concerns training effectiveness requirements for flight simulators. He received his commission in the Air Force in 1982 with a BS in Experimental Psychology/Human Factors from the Air Force Academy. He has a MA from Dayton University in Experimental Psychology.

SIMULATION OF THE GROUND COMBAT ENVIRONMENT

Allen T. Irwin
Science Applications International Corporation
3045 Technology Boulevard
Orlando, Florida 32826

ABSTRACT

Modern weapon systems, as exemplified by the M1 tank and the AH-64 attack helicopter, are placing new demands on simulators. Because of the increased costs and hazards associated with training using actual equipment, a demand is emerging for combat simulators that can train tactical commanders and crews to operate and fight in the ground combat environment. This paper examines the background of combat simulation, the specific requirements generated by the emerging demand, the present technical capabilities to support such requirements, and a brief look at the future growth needs of the technology.

INTRODUCTION

In 1415, at the battle of Agincourt, the flower of French Knighthood elected to charge the badly outnumbered English troops of Henry V. Unfortunately for the French, the English had selected defensive positions on high ground that the French could only reach by first crossing a low lying swampy area.

Under the weight of their heavy armor and equipment the French horses quickly became bogged down in the mud. The French charge fragmented and slowed. Some elements were stopped all together. Many knights were unhorsed without any enemy action at all. The English were not just watching. The immobilized French became easy pickings for the feared English long bow. As a result, the French were never able to deliver the heavy cavalry charge that would have rolled up the English lines and caused the defeat that the English had feared as the battle began.

In 1775, Colonists, firing from covered positions behind trees and fences, turned what had been a tactical defeat at Concord Bridge into a near rout of the apparently victorious British forces.

In the Civil War, time and time again, the terrain determined where battles were fought, who fought them, how they were fought, and who the victors were.

Custer, attempting to deal with the terrain, split an already small force, and won glory (but nothing else) at the Little Big Horn.

World War I saw the introduction of the machine gun, which, once and for all, put an end to charges of massed troops across open terrain. The airplane meant that terrain had to be used to hide troops from aerial observation and bombardment as well. The tank, although immune to the machine gun, still had to cope with the ground it was required to cross. Antitank ditches and obstacles entered the ground environment of combat.

World War II, Korea, Vietnam, the Middle East, Afghanistan, all have served to reinforce the importance of terrain to land combat. Modern technology only extends the lesson. "If you can see it, you can hit it. - - - If you can hit it, you can kill it." The day of massive formations, moving ponderously against an objective in spite of the intervening terrain, is long gone. At the battle captain level, use of terrain is a critical skill. For crews of weapon systems employed on the cutting edge of the land battle, terrain, if properly used, will allow the achievement of the multiplication factors that our technology offers. Failure to make use of the terrain will negate those factors.

As simulators move into the tasks of training our land forces for combat, representation of terrain becomes a matter of vital importance. Specific skills may be taught in isolation, but for tank crews, cavalry scouts, infantry squads, artillery FIST teams, armed helicopter crews, and their immediate commanders, tactics equate to terrain utilization. If a simulator cannot teach terrain utilization, it cannot teach tactics. If it ignores terrain, then the chances of negative training are multiplied.

BACKGROUND OF GROUND COMBAT ENVIRONMENT SIMULATION

Three different applications, each with their own objectives, have established the current state of the art for simulation of the ground combat environment. Analytical simulations have developed the capability to evaluate the environmental effects on weapon systems and sensors. Exercise Simulators have been able to provide a real-time representation of the environment from the point of view of map readers, and Flight Simulators have developed the ability to picture the ground environment for pilots flying over it.

ANALYTICAL SIMULATIONS

Analytical simulations have been used for years to evaluate the effectiveness of various combat systems. Such models do not necessarily run in real time but are used to make repeated trials of varying mixes of forces and equipment against various threat scenarios. To be acceptable, such models are required to deal with the effects of the ground environment. Sensors require a line of sight to their target if they are to detect it. Direct fire weapons also require a line of sight to their targets if they are to engage them. To determine if a line of sight exists between two points, the model must evaluate the shape of the intervening terrain, vegetation, objects such as houses or other structures, and earth curvature.

If discrete representation of the earth surface is attempted, certain problems become immediately apparent. The most obvious problem is that if the target is a crawling man, and the sensor is at ground level (as are most ground combat sensors), then any intervening terrain feature the size of the target is able to prevent the target's detection. In other words, variations in the terrain on the order of 12 inches become significant in the deterministic evaluation of the environmental effects of terrain on sensor performance. Given that a battlefield environment of sufficient size to allow significant modeling will be at least several kilometers on a side, it quickly becomes apparent that an attempt to represent terrain variations on the order of 12 inches is a gargantuan task. When the need is to discretely model all vegetation that can impact sensor performance, it becomes obvious that the approach is impractical. As a result of this difficulty, analytical models that must deal with the ground environment generally use a sampling or statistical representation to avoid the masses of data that would otherwise be needed.

One of the most common methods of sampling is to store environmental data for intersecting points of a grid (see Figure 1). For example, terrain data can be stored for points that are spaced at 50-meter intervals. This would require that 400 points be stored for each square kilometer. Many kinds of information need to be stored for each of these points: elevation, gradient, soil type, surface condition, hydrology, vegetation, cultural objects, etc. Means of encoding this data allow reduction of such information to approximately 15 8-bit bytes. Thus, the data to represent a 10-kilometer square area would require 400 x 100 x 15 or 600 kilobytes. Such a database would not be a problem for the manipulation capabilities of a modern computer. But let's examine some of the problems of such sampling. The problems generally come down to that which is missed in the sampling process. For example, if a 20-meter-wide antitank obstacle is placed into the database, each row of sample points that crosses the obstacle will only have a 40% probability of recognizing the presence of the obstacle. As a result, interrogation of the database will only indicate occasional fragments of the obstacle and it will be logically impossible to determine if the samples indicate a single, unbreached obstacle, or several obstacles with gaps in between. Road nets become equally undefined with such sampling frequency. It becomes impossible to establish connectivity within a net so represented.

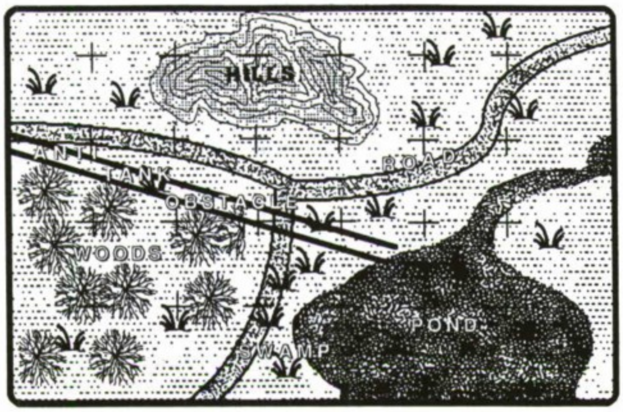


FIGURE 1. GRIDDED SAMPLING OF TERRAIN

The obvious answer is to increase the frequency of the sampling process. If the frequency is doubled (to every 25 meters) there is still a frequent loss of data on roads and other linear features. If a 12.5-meter interval is used, most major roads will be picked up, but trails, single-lane roads, and streams will still be missed a significant amount of the time. Meanwhile, the fourfold increase in linear sampling frequency has generated a 16-fold increase in data storage requirements. The 10 x 10 kilometer gaming area now requires 9.6 megabytes of memory. Computational requirements likewise have increased, and the examination of the terrain between two tactical units that each occupy a front of 200 meters, and are separated by a kilometer, would require the accessing of 1,280 data points. Not only would that number of data points be required, but the determination of whether a line of sight existed between the two units would require the examination of multiple combinations of the data points.

Another approach used in analytical models avoids some of the cumbersomeness of data representation by defining the terrain as consisting of various areas of homogeneous terrain (see Figure 2). The various characteristics of each area are then treated as constant throughout each specific area. This technique is often combined with a stochastic rather than deterministic means of defining the characteristics of terrain. Thus, in any area,

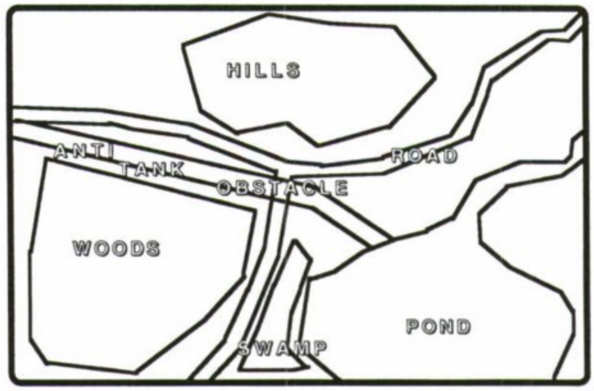


FIGURE 2. TERRAIN AREAS

the operation of a sensor may be described as a probability distribution function about a value representing the average detection range for a given target type. Values for trafficability can be assigned to a given area and any type of terrain-related characteristic can be specified in the same manner. The fidelity of the model then depends on the skill with which areas are defined and placed. Large numbers of areas, carefully placed, will give good performance. If the areas are made small, then they can be given height and slope parameters and can represent the shape of the terrain in a discrete form rather than only as shape affects the other parameters. When made small enough, the areas can be made equal-sized regular polygons of either 3, 4 or 6 sides. Again, the degree of representation depends on the resources devoted to the task in terms of database size, model speed and hardware capacities, and the effort required to prepare a database.

EXERCISE MODELS

Exercise Models are those simulations that are meant to support training exercises such as Command Post Exercises. Army Training Battle Simulation System (ARTBASS) and its predecessor Combined Arms Tactical Training Simulator (CATTS) are examples of such simulations as is the Joint Exercise Simulation System (JESS).

Their purpose is to provide a realistic setting for a command team to exercise its ability to plan for and control the execution of a battle. The first two models are used for Battalion level exercises, but the approach can be generalized to almost any level of command post. JESS, for example, is used for Corps level exercises. The technique utilized in all three cases is to interface human role players with a computer simulation of the battle (see Figure 3). The role players then provide the communications that would be the Command Post's view of the battle. Upon receipt of orders from the Exercise participants, the role players update the computer simulation

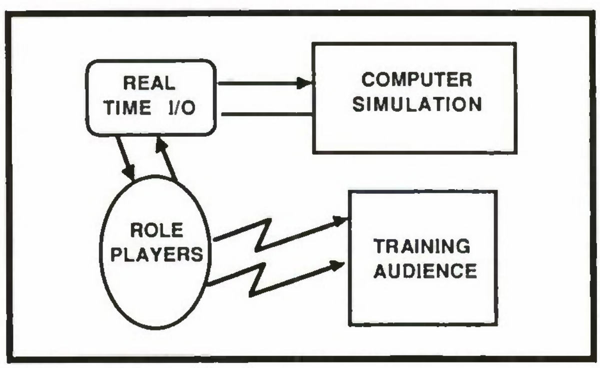


FIGURE 3. EXERCISE MODEL

accordingly. Such models are similar to the analytical models described above, except that they must operate in synchronization with real time, accept modifications during operation, and provide a wealth of data to role players in an interactive mode. These applications must, of course, make certain compromises to meet the additional requirements that are not imposed on normal analytical models. As an example, ARTBASS and CATTS use a digital terrain database that is derived from data supplied by the Oefense Mapping Agency. The data is in the form of a gridded sampling of the area of interest. JESS used a terrain database composed of regular hexagonal areas approximately 3 kilometers across. Using this information, the simulations calculate intervisibility between units, fields of fire, movement rates, obstacle crossings, and other terrain-related parameters. The information is displayed to the role players as symbology superimposed on a color map background. In ARTBASS, the capability is also provided for display of a three-dimensional projection of the terrain to aid in role-player analysis of the tactical situation. All of the calculations of the simulation are performed at a rate that allows the results to keep pace with the actual clock time of the battle.

FLIGHT SIMULATORS

Flight simulators have approached the representation of the ground environment from a different perspective -- literally. The principal requirement for terrain modeling for flight simulators has been the need to provide the visual stimulation that student pilots need to correctly navigate and orient their aircraft. As a result, the emphasis has been on pictorial content and correct simulation of motion. Thus, the visual scene is updated at 30 to 60 Hertz. Initially, high-flying aircraft needed only abstract patterns to portray the surface of the earth, and the only pictorial representation of objects was of the airfield environment. As technology has matured, the use of helicopter flight simulators and close support aircraft trainers has led to the development of detailed Computer Generated Imagery (CGI) of the terrain of the battlefield. Only recently has the special purpose computing hardware needed to produce detailed scenes in the time required become available. Prior to that time, techniques such as TV cameras and scale model boards were used when detailed scenery was required. Such methods are still being used, however, their primary limitation, the fact that the terrain data is not available to computational models, makes them less desirable as CGI systems become able to compete in terms of image quality.

In a CGI database, terrain shape, objects such as trees and houses, and features such as roads and rivers, are all represented as some collection of polygons. Each polygon has certain data associated with it to locate and describe it. In early data-base applications the only data required was color. Now it is often necessary to tag a polygon with an object type code and other information as well. In the most modern systems, polygons can also be associated with data needed to select a texture pattern that provides enhanced visual realism. Other systems are being developed that manipulate recorded images to produce highly realistic visual pictures. Of course, even with the most advanced hardware, processing any data still requires finite lengths of time. Thus, trade-offs are needed to keep the quantity of information used for modeling down to the amount that can be handled in the update time available between visual images. In many CGI systems that are still in use today, this has resulted in what are described as "lollypop trees and billboard mountains" (see Figure 4). As systems now in development emerge, the visual affects will be vastly improved. However, some of the effects are being achieved by techniques that improve only the visual representation, not necessarily the internal model. As an example, texture effects can be used to cause a two-

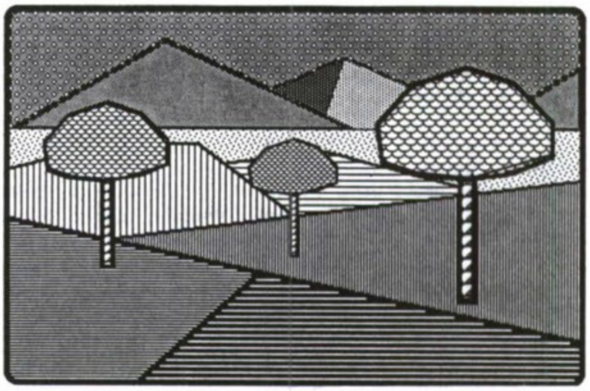


FIGURE 4. SIMPLE COMPUTER GENERATED IMAGE

dimensional model of mountains to take on the appearance of depth, producing apparent hills and valleys that appear very real. But, if one interrogates the model to determine the range to a particular hill and to the adjacent valley, the range will be the same because the shape representation is only a visual effect.

Thus, while visual systems used with flight simulators represent an increase in detail over most models used for other purposes, they have, in general, made trade-offs in favor of visual effects in order to overcome capacity limitations imposed by the high update rate requirements inherent in such applications.

EMERGING REQUIREMENTS

Two specific Army training devices, both in production, highlight what can be identified as a new level of simulation requirement that is emerging as a major demand for future simulation applications. One of these devices, the Conduct of Fire Trainer (COFT), is intended to teach tank crews and fighting vehicle crews the gunnery skills necessary to survive in combat. The other, the Attack Helicopter Combat Mission Simulator (CMS), is even more ambitious, seeking to train not only gunnery skills, but combat maneuvers in the Nap-of-the-Earth environment as well. Both devices use visual systems of the sort associated with flight simulators. Both add to them the ability to portray a variety of moving targets that represent infantry, tanks, trucks, and helicopters. The targets are given the ability to shoot at the craws under training, and given inadequate performance, a simulated kill can be scored against the trainees. Thus, both trainers simulate the ground combat environment to a degree that has not been done before.

These devices are intended to train the battle crews that operate in direct contact with the enemy and the environment. Their war is vastly different from the arena of maps and symbols. Their actions are much closer to the hip-shoot style of the gun fighter than they are to the analytical problem solving style of the chess player.

It is evident that the requirement for such simulators will not only increase in quantity, but also in the demand for performance. The nature of modern, high-technology weapons is such that it becomes increasingly difficult to train for combat without incurring major costs and safety limitations. The modern tank fires a round from its main gun that if fired at optimum elevation would come down over 43 miles away.* Along that

*Firing 105 mm Cartridge APFSOS-T, M774, as shown in Change 2 to FT 105-A-3, dated 1 August 1982, Headquarters, Department of the Army, Washington, O. C.

trajectory the projectile would reach an altitude of more than 19 miles or more than 100,000 feet. To allow the firing of such ammunition, or even less powerful training rounds, requires the clearing of large impact areas, and the imposition of range safety procedures that practically eliminate any possibility of tactical realism from the training. In addition, the costs of such rounds, the wear and tear on the gun and vehicle, and the costs of fuel, yield operating expenses that greatly restrict the possibility of live training.

The attack helicopter is even more constrained. To all the problems cited for the training of tank crews must be added the problem of flight safety. As a result, there are portions of the mission that cannot be practiced in a live environment.

These training constraints mean that a major training shortfall exists for combat vehicle crews, aircrews, platoon leaders, and company commanders (see Figure 5). It is this shortfall which is now producing a demand for simulators to meet the training requirements of combat training in the ground environment.

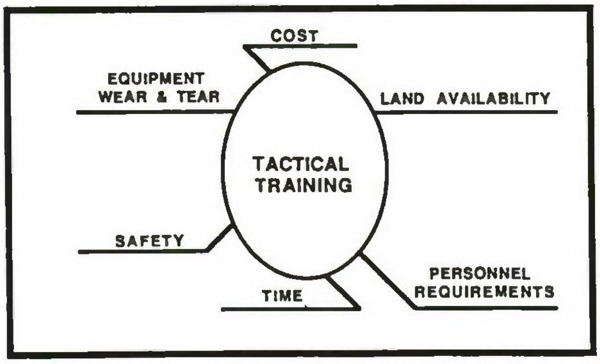


FIGURE 5. CONSTRAINTS ON TACTICAL TRAINING

GROUND COMBAT ENVIRONMENT SIMULATION REQUIREMENTS

If simulators are to be used to teach ground combat tactics at the Company level and below, then certain specific tasks or activities must be supported. These tasks and how each is affected by the Ground Combat Environment are all examined in the following subparagraphs to determine what simulation requirements result.

SITUATION ASSESSMENT

A tactical commander assesses the situation in terms of Mission, Enemy, Troops, Terrain and Time Available (see Figure 6). His mission will be to attack, defend, or delay. His situation assessment will influence how he evaluates all other factors.

The enemy, or what is known about him, is of major importance. In a combat simulation, the representation of the enemy is of equal significance. The complicating factor is that in the majority of cases, the enemy will be detectable to the ground combat commander by visual means only. Therefore, a significant requirement for simulation is the ability to portray the enemy visually in a realistic manner and in suitable numbers and variety. Not only must the enemy and his equipment be simulated, but so must the effects of his equipment. One of the most drastic effects on the battlefield is artillery fire, yet it has not been included in the CMS. It is too demanding of simulator resources and has been traded away. Artillery fire is of significance to the tactical commander not only because it is a threat, but because it is an indicator of the enemy's intent.

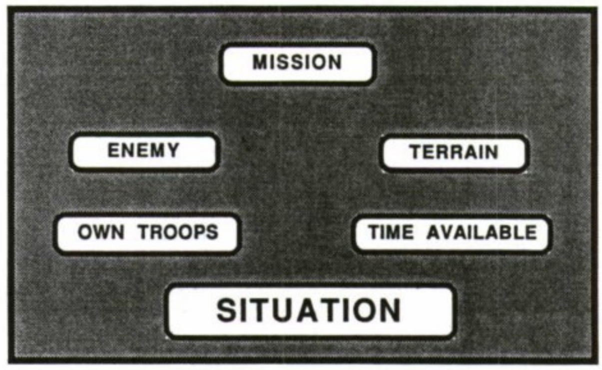


FIGURE 6. SITUATION ASSESSMENT

Friendly troops are the next element of the commander's assessment. Not only does he consider his own troops, but also the supporting and adjacent troops that will affect his capabilities. In most simulators today, the portrayal of friendly troops is rudimentary at best. To adequately train a commander in the constant situation assessment process that allows him to make the necessary decisions in combat, requires the full simulation of friendly troops and the effects of their weapons in the battle area. One of the most common mistakes made by inexperienced commanders in battle is the failure to make use of all the support resources available (e.g., mortars and artillery fire). If the simulator training him cannot represent such resources, such mistakes will be reinforced.

Terrain is the next element of evaluation (see Figure 7). It is of such importance to the tactical commander that it is further analyzed in terms of Cover, Observation and Fields of Fire, Concealment, Obstacles, Key Terrain and Avenues of Approach.

Cover is the ability of terrain features to protect troops and equipment from weapons effects. Given the deadly effects of modern weapons, survival on the battlefield depends on the maximum use of available cover. The ability to identify covered positions and routes will be a key skill for anyone on the tactical

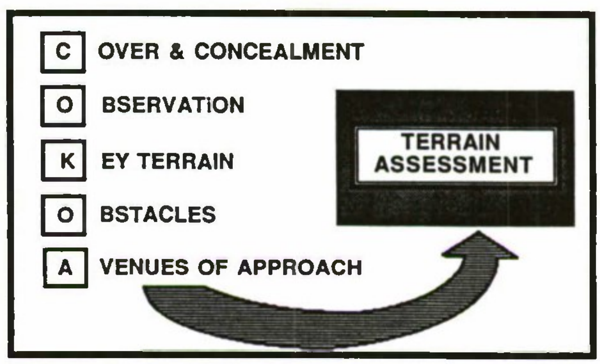


FIGURE 7. TERRAIN ASSESSMENT

battlefield. A key requirement for simulation will be the ability to portray terrain such that cover is available and can be recognized and utilized.

The eyeball is still the chief sensor on the tactical battlefield; therefore, the ability to find terrain that allows observation of the battlefield is also critical. Good observation generally means good fields of fire. If you can't see it, you can seldom shoot it. This also means that if you are not seen, you are less likely to be shot at.

The obverse of observation is concealment. The

simuleted environment must allow the tacticel treinee to leern to meke use of bettlefield conceelment. Note thet while cover is elso conceelment, conceelment does not elweys provide cover.

Obstacles ere those things which prevent movement end cen be naturel or men mede. Exemples ere rivers, cliffs, dense forests, swamps, mine fields, tenk treps, ditches, wells, end rubble. Not only must such obstecles be present in the simuletion, but their representetion must be complete, not merely cosmetic. In meny visuel simuletions, e river is only different from surrounding terrein in the fect that it is blue. For tacticel simuletion, it will be necessery to determine depth, current speed, bottom condition, benk condition, etc., for these ere the factors thet decide if men end vehicles can either ford or swim a river. It should be noted that this levies en edditionel requirement on visuel systems to provide the visual cues that allow the eveluation of the ebove fectors. For vehicle end foot movement, the besic surfece condition, soil type, vegetation, end slope will determine if a piece of ground is or is not en obstacle. Such fectors must be present end discernable in the simulation.

Key terrein is any terrein feeture, the possession or control of which by one side or the other, cen influence the outcome of the battle. Exemples ere high ground thet offers good observation end fields of fire, end defiles or pesses thet cenelize movement, etc.

Consideration of the preceding fectors allows the tecticel commender to identify evenues of approech either leading into his own position or into the enemy position thet is his objective. It is the avenues of approech that shape defensive end offensive possibilities end simuletion of the environment must meke them discerneble.

The commender must elso consider the time eveileble in selecting his course of ection. Time pressure is one of the key feetures for tecticel treining. Reel time performence is criticel to eny tecticel treiner.

TARGET ACQUISITION/DETECTION/IDENTIFICATION/PRIORITIZATION

These tasks are somewhet specific to perticuler weepon systems. However, certein common threeds pertein to ell. On the tecticel bettlefield, these tasks ere generelly performed based on information obteined visuelly, either with or without eids such es binoculers, optical sights, FLIR, or TV. This meens thet the essentiel feeture of simuletion for this tesk will be visuel representation not only of the threat forces, but of ell the elements such es terrein shepe, vegetation, buildings, smoke, heze, lighting, shedow, dust, end weepons effects thet influence the performence of these tesks.

TARGET ENGAGEMENT/DISENGAGEMENT

These tesks ere egein peculier to the weepons system in consideretion, however, treining in these tesks levies e need for correct simuletion of threet response. The ections of en enemy tenk when teken under fire ere likely to be quite different from e moving terget on e renge thet just continues sedetely elong its wey. It is the portreyel of this difference in behevior thet seperetes e tecticel treiner from e gunnery treiner.

The intelligence of the threet model will be e mejor consideration in tecticel treining. Threet response to engegement must be coordineted. Not only will the tenk being shot et reect, but so will all enemy elements eveileble. This might include ertillery end eir strikes es well es return fire from the engeged formation. The enemy response must include meneuver es well es fire. Our fire discloses our position end will influence the ections teken by the threat forces. Under some conditions he mey seek cover end withdrew. Under others, he mey ettempt to close end overrun friendly positions. Agein, the ections of friendly forces must be included in the simulation, for no one element ects in isoletion.

MOVEMENT AND POSITION

Most tecticel plens begin with meps. Objectives, defense positions, end routes of movement ere often selected from meps. As e metter of fect, most tecticel commenders ere given their positions end routes from orders implementing mep-besed plens. From this, two specific skills emerge es necessities for the tecticel commender, be he in e tank, e helicopter, e fighting vehicle, or on foot. The first is the very besic skill of mep reeding. Not just interpreting the mep symbols end colors; that is eesily leerned. The reel requirement is the ebility to correlete the map with the terrein by observation, the ebility to determine thet e specific feeture in view corresponds with e specific feeture on the mep. From that comes the ebility to extrepolete the terrein not in view from the symbology on the mep. The besic treining in this skill cen be teught on the ground; however, eny tecticel simuletor must support the use of this skill if it is to be extended to the tecticel environment.

The second requirement, end perheps the most importent of ell, cen be called terrein exploitetion. Everybody on the tecticel bettlefield must know how to gein meximum edventege from the use of the terrein in eccomplishing his mission. The unit thet does not teke maximum edventege of cover end conceelment will teke heevy cesuelties. The unit thet does not control dominent terrein will be domineted by the enemy. The unit that does not recognize obstacles to movement will be trepped egeinst those obstecles by the enemy. On the high-technology bettiefield of modern werfere such errors will be penalized by repid destruction.

It is imperetive, therefore, thet the ebility to trein such skills be regeined. Our modern equipment hes robbed us of the freedom to conduct reelistic tecticel treining. No longer cen e small unit commender teke his unit to e convenient locel treining erea end expect to conduct reelistic training in tecticel employment. The cost is too high, too much lend is required, end safety consideretions ere prohibitive.

The requirement thet this training need pleces on simuletors is to present to the student sufficient deteil of the terrein to ellow the exercise of tecticel decision-meking skills. The presentetion of the deteils cennot be only visuel. The entire simuletion must respond to the terrein. If en ettempt is mede to ford en unfordeble river, the tenks cennot be ellowed to drive right ecross the surfece es if it were peved with concrete. If e student ettempts to negotiete e swemp, progress of the vehicle must be simuleted correctly. All fecets of the terrein must be eveileble in e coherent menner to ell elements of the simuletion. An enemy vehicle must respond to the terrein with equel fidelity.

EFFECTIVE TECHNIQUES FOR MANAGING ENVIRONMENTAL DATA

As described in the earlier sections of this peper, the beckground of Ground Combet Simuletion is twofold. One set of encestors cen be described es the enelyticel or nonvisuel simuletions, while the other set is the flight treiner femily of simuletors featuring eleborete visuel representetions of the ground environment. This twofold source of development hes creeted two sets of technicel cepebilities: The enelyticel models heve the ebility to hendle terrein enelysis in terms of cross-country mobility, intervisibility of tecticel units, etc. They do not heve the

ability to present the detailed, real-time, high update rate visual scene that is the specialty of the flight simulators. The flight simulator visual systems are just that. They have been designed specifically to present visual images. To achieve their speed, they employ special purpose hardware, capable of the high throughput speeds needed to generate visual imagery from a large, complex data base. However, the hardware has been designed to support image generation and not information retrieval. Thus, the visual systems have only a limited capability to support other modeling functions. The data bases which contain the detailed information representing the ground environment are too massive for software manipulation in the time frames required. (It should also be noted that only the most elaborate of visual data bases begin to approach the level of detail needed for tactical training. Nor do most of these data bases contain information such as hydrological data on rivers and soil condition data needed to support ground movement calculations.)

In systems that are going to require highly detailed visual terrain presentations, with full freedom of motion through the data base, it is to be expected that the high resolution CGI visual system will be the implementation of choice. While recorded video can produce highly realistic terrain, it can only present the pictures that have been recorded, therefore, even with the electronic manipulation that is being exploited for special effects, free movement through the terrain data base is not supported. CGI systems are taxed, however, by the demands made by present day systems for data concerning the terrain. Designed to support the visual presentation, the data transfer required to support the modeling functions are essentially add-ons in the design. Furthermore, it is not necessarily desirable to alter the design of such systems to require them to support the emerging data needs. It is best that the resources of such systems remain primarily dedicated to the visual tasks.

The alternative to even larger capacity visual systems capable of doing all things for all functions is a data base for terrain data that is dedicated to the nonvisual functions. Necessarily, the nonvisual and visual terrain data bases must have a high degree of correlation. However, by careful division of the tasks that are assigned to each data base, the tasks assigned can be accomplished by both without creating anomalies. In fact, the nonvisual system terrain data base should be capable of implementation in a general purpose processor (as opposed to the special purpose hardware that is required for the visual system).

The key to the division of tasks for such an implementation is to observe two guidelines:

1) Leave to the visual system those tasks which are most likely to create visual anomalies.

2) Minimize the data that must be handled at any one time by the non-visual system data base.

Thus, determining the strike of a projectile should remain with the visual system because of the requirement for almost perfect correlation with observed events. The determination of vehicle attitudes and motion (for ground vehicles), however, requires very localized terrain data with low update rates; therefore, it is feasible to assign this task to the nonvisual system data base processor. Many other factors are drawn from the effects of terrain, but do not require direct equivalency with the visual scene. For example, if it is necessary to assess casualties against a unit that is simulated as part of the trainer, the data representing the moderating effects of the terrain (i.e., cover and concealment provided by the surroundings) can be stored for general regions of the gaming area without reference to the polygons that make up the visual scene.

It should be possible to provide local terrain data for simulated units (targets, friendly troops, ownvehicle) in a manner that supports all but the most visually critical phenomena using a general purpose processor acting on a structured data base containing all necessary data in great detail. The key is the selective retrieval and update of local areas of interest for each of the simulated elements active in the combat environment.

Within the nonvisual terrain data base, the structure of data can be designed to support the specific needs of information. Not all data needs to be stored in the same resolution, so it is possible that multiple structures may be employed. Intervisibility data may be stored by region, while slope data may be stored by a sampling grid. Trafficability may also be stored for large regions.

Another technique that can facilitate interaction with such a data base is the concept of nested regions. In such a structure, the terrain is divided into large regions of common features, and parametric data is stored for such regions. Each region in turn is divided into smaller regions each of which have data stored that represents finer detail. The subdivision of regions would continue until very high-resolution data is stored for localized positions. In using such a structure, the model would retrieve and use data for the largest region feasible, going into deeper subdivisions only as needed. For example, if it was desired to determine if a mechanized infantry battalion would detect a simulated enemy regiment, detection parameters for a region the size of the operating area of the battalion would be used. Only as the battalion moved into a new area would it be necessary to update the parameters. If, however, the modeling of a particular company within the battalion were being accomplished, then the next level of detail would be accessed, however, only for the area of interest of the company in question. If it was necessary to model the motion of the company commander's vehicle, then the data for the area of interest for that vehicle would be retrieved and used. But in each case, only the parameters that require high resolution would be taken from the more detailed subdivision. Thus, even for the company commander's vehicle, the determination of continued radio contact with the battalion command post would be determined based on parameters for the battalion level area of interest.

Such a design allows the quantity of information that is stored for the lowest levels to be minimized, and also lowers the update burden by decreasing the overall frequency with which data must be updated.

To further alleviate the processing involved, care must be taken to see that information about the environment is stored in the most usable form. As an example, if point elevation data is retrieved for the area between two simulated units, a prediction can be made as to the probability that they see each other. Because the retrieved data is a set of samples, the prediction will never be an absolutely deterministic answer, but rather a statistical answer. Therefore, it is equally effective to store a single parameter representing intervisibility for the area. Not only is less access to the database required, but the computation once the data is retrieved is also minimized.

No discussion of ground combat simulation would be complete today without some consideration of the DARPA SIMNET project. SIMNET has chosen to leave the terrain data base in the CGI system for the M1 Tank simulation, with necessary data being passed to the model computer: (1) In the Managemant, Command and Control (MCC) system, however, a terrain data base is used by the host computer. This is necessary because the MCC does not utilize a visual system. (2) Thus, SIMNET uses a variation on the approach described above. It addresses many of the requirements for ground combat environment simulation that are identified above. Its achievements

have impressed most observers. It is clearly unique in its bringing together of many technological opportunities and applying them to the problem of ground combat simulation. However, another important feature of the SIMNET experiment must be remembered. SIMNET is also a test of the concept of "selective fidelity." To achieve the performance that it has demonstrated, it has necessarily made many compromises in terms of fidelity. Some of the effects of these compromises have been identified already, as when it was recognized that the 125-meter spacing of the terrain grid tended to smooth out defilade positions for tanks. Over specification of fidelity requirements for simulators is a cost driver, and hopefully, SIMNET experimentation will shed significant light on what level of fidelity is, in fact, training effective for which tasks.

FUTURE EXPECTATIONS

The training needs described above will not go away. If anything they will intensify. We can expect, therefore, that technology will emerge to satisfy these needs. Because of the quantity of data involved, we can expect that special purpose hardware will continue to be the basis of computer generated imagery. To satisfy the additional requirements, more data will have to be present in the data base. This probably means that additional hardware capabilities will be required to allow access to this additional data. The solution is not necessarily limited to the realm of visual data base expansion. Hybrid solutions are conceivable such as retrieval of keys from the visual data base allowing software access of the necessary additional data. It is also possible that a preprocessor could set up software-accessible data from the visual data base while still retaining the necessary one to one correspondence.

Data storage and retrieval technology will continue to ameliorate this problem. Distributed processing techniques and the continuing reduction in hardware costs will also contribute to eventual solutions.

CONCLUSIONS

A clear need is emerging to pick up more of the tactical training burden using simulation technology. The need will strengthen as new systems are introduced to the ground forces that increase the cost and difficulty of conducting tactical training using actual equipment.

Full and consistent representation of the Ground Combat Environment is essential to tactical training. The environment includes all significant terrain features, cultural objects, road networks, and enemy and friendly forces.

Current simulation technology is only marginally able to support tactical training. The full requirement of ground tactical training will require the simulator industry to add capabilities beyond the current technology.

The limitations of current simulator technology must be considered when specifying trainer requirements. Alternate means of training must be used to accomplish objectives that cannot be met in the simulator. As new technology emerges, simulators can be enhanced to provide a more complete training environment.

The first step in meeting the requirement for tactical simulation is recognition of the impact of the emerging requirements on simulation technology.

REFERENCES

(1) SIMNET M1 ABRAMS MAIN BATTLE TANK SIMULATION, Software Description and Documentation, Report No. 6323, 12 February 1987; Chung, James W.; Dickens, Alan R.; O'Toole, Brian P.; and Chiang, Carol J.; BBN Laboratories Incorporated for Defense Advanced Research Projects Agency (DARPA)

(2) The SIMNET Management, Command and Control System, Report No. 6473, 31 March 1987; Pope, Arthur R.; Langevin, Tim; and Tosswill, Andrew; BBN Laboratories Incorporated for Defense Advanced Research Projects Agency (DARPA)

ABOUT THE AUTHOR

Mr. Allen T. Irwin is a Vice President of Science Applications International Corporation (SAIC), and is currently assigned as a Project Director at the Orlando facility. He has provided technical support to PM TRADE and NTEC for over 9 years. His previous experience includes software acquisition support to other Army Projects under PM ARTADS, development and maintenance of simulation software for the U. S. Air Force, and active service as an Armor Officer in the U. S. Army.

THE SIMNET VISUAL SYSTEM

Richard S. Johnston
BBN Delta Graphics Inc.
Bellevue, Washington

ABSTRACT

The SIMNET System, developed by the Defense Advanced Research Projects Agency, allows for collective team training of military personnel. Using a network of multiple simulators, the initially fielded system trains armored vehicle crews in the land battle environment. Trainees engage in two-sided, free-play, tactical exercises on terrain matching real world locations.

The SIMNET program necessitated developing a new visual system. This system required a sufficient number of independent viewports for full crew training, a large number of various moving models, and a database capable of providing adequate detail. Also, these requirements had to be met with an extremely low cost device.

A visual system with eight independent viewing channels was designed. Each can display up to 1000 visible, four-sided, textured, antialiased polygons at a 15Hz rate. Using a hybrid depth buffer architecture, the program can process over 150 moving models. Pixel throughput meets the higher depth complexity typical of ground based simulation. Dynamic database techniques allow training over large areas. And, automated database generation software rapidly creates databases which closely match real locations.

INTRODUCTION

During the fall of 1984 Delta Graphics Inc. was contracted to design and build a visual system for the Defense Advanced Research Projects Agency's SIMNET system. This paper will describe the requirements and design of the computer image generation system developed to meet DARPA's goal of an effective, low cost visual training system.

The paper will begin with a brief explanation of the SIMNET concept and its visual system requirements. Following this, an overview of the visual system architecture will be given. Finally, key concepts in the visual system design will be examined in greater detail.

In a conventional simulation system the student learns to perform a series of tasks, such as those needed to takeoff and land an aircraft, or to fire a tank's weapon systems. The student is practicing against preprogrammed scenarios under the direction of an instructor. SIMNET, on the other hand, is a large scale combat simulation network that allows collective team training of military personnel.

In the SIMNET system individual simulators are networked together. The trainees are divided into teams and allowed to engage in a two-sided, free-play, tactical exercise. Crews are competing against real people who can think and make decisions based on battlefield situations.

The initially fielded SIMNET system is used to train armored vehicle crews in a land battle environment and allows hundreds of simulators to be networked together. Each simulator is configured to represent a single vehicle, such as an M-1 tank.

To allow realistic battlefield interactions, a full complement of crew positions must be provided for. This leads to a large number of vision blocks in each simulator. With the large numbers of simulators needed, the cost per simulator, and thus per vision block, must be minimized. This fact was an overriding goal during the design of the SIMNET visual system and leads directly to the idea of selected fidelity.

The design goal when using selective fidelity is not to simulate the world exactly, as yet an impossible task even with unlimited money, but to simulate only what is necessary to train the desired task. For example, in SIMNET's M-1 simulator not all switches are operational. Some switches, such as the bilge pump, are painted on as they were deemed unnecessary in training team combat tactics.

An example of this in the visual system is the image update rate. Instead of updating the image at 25 to 60 times per second, as is standard in most visual simulators, the SIMNET visual system updates the image 15 times per second. It was decided that, after analyzing the maximum velocities and turret slew rates of the M-1 tank, this update rate was sufficient to train ground based personnel.

SYSTEM SPECIFICATIONS

The unique requirements of SIMNET brought about the need for a new visual system. The specifications for this visual system were developed with the objectives of the total training system in mind. This section of the paper will examine the requiremetns of SIMNET in relation to the visual system specifications of the initially fielded M-1 ground based trainer. The primary goal of SIMNET is to provide an effective training system at the lowest possible cost. A summary of the visual system specifications is shown in Table 1.

PARAMETER	CIG SYSTEM
Independent Viewing Channels	8
Occulting Levels	524,288
Frame Update Rate	15 per second per channel
Transport Delay (at 15 Hz)	<150 milliseconds
Computed Screen Resolution	320 x 128, 200 x 200
Displayed Screen Resolution	640 x 256, 400 x 400
Video Format	RS-170 RGB
Field-of-View (FOV)	Frame to frame selectable
Potentially Visible Polygons per Frame	1000 polygons/channel
Online Database Storage Capacity	70 million bytes
Active Area Memory	1.5 million bytes
Terrain Grid Spacing	125 meters
Depth Complexity	3.8 at 15Hz.
Color Resolution	4096 colors
Anti-aliasing	Yes
Distance Fading	Yes
Texture Generation With Transparency	Yes, 16 transparency levels
Stamp and Perspective Texturing	Yes
Model Level-of-Detail Control	Yes
Moving Models Per Gaming Area	150
Texture Patterns	Up to 256
Laser Range Finding	Yes
Database Size	Up to 3.75 million polygons

SIMNET SPECIFICATIONS
TABLE 1

In the SIMNET world hundreds of combatants exist simultaneously on the battlefield. Each of the combatants must be able to see all others, both friendly and enemy. In addition, many support vehicles, such as fuel and ammunition trucks, must be visible and movable. This supports the requirement for a system that can handle large numbers of moving models.

Because the orientation of a tank's turret and gun is important (Is that gun pointing at me?), multiple dynamic coordinate systems must be supported for each model. To properly use cover and concealment techniques, the vehicles must be free to maneuver anywhere in the environment in the same way as a real tank. Numerous special effects may also exist at any time. These include ground and air bursts fired from vehicles in the simulation or from remote artillery, burning and damaged vehicles, and dust clouds from moving vehicles. To meet these requirements the visual system allows for the display of over 150 simultaneously moving models and 450 dynamic coordinate systems. Each of these is updated every frame.

An M-1 tank is occupied by four crew members; the gunner, loader, driver, and commander. It was determined that eight viewports were needed: one for the loader, corresponding to the loader's periscope; three for the driver, corresponding to the three periscopes used while driving in the closed hatch mode; and three for the commander which can rotate independent of the turret and correspond to the views from the commander's hatch. The above seven views all have aspect ratios of 2.5 to 1 as they represent the slit-like views available in the M-1. The gunner has a view corresponding to the one visible through the gunner's primary sight. This channel has an aspect ratio of 1 to 1 and is also viewable by the commander on a separate monitor.

Terrain and cultural features on the terrain must be modeled with enough detail to allow crews to use normal battlefield techniques. These include cover and concealment, and hull and turret defilade positioning. The effects of varying terrain slopes on tank movement is also required. In addition, providing for training in terrain that matches real world locations, as closely as possible, is desirable.

To meet these goals, the visual system needed sufficient polygon throughput to model terrain generated from defense Mapping Agency Digital Terrain Elevation Data (DTED). The SIMNET system uses DTED data resampled to a 125 meter grid spacing. A technique known as terrain relaxation is used in areas of constant slope to reduce polygon counts. Greater terrain detail may be added by using microterrain where needed.

It was determined that a polygon throughput of 1000 potentially visible, four-sided, textured polygons per frame per channel was sufficient to meet the SIMNET requirement. A potentially visible polygon is a front facing polygon, after clipping, that is sent to the screen. These polygons may be occulted by other closer polygons. Of the total 1000 polygons, approximately 300 are dedicated to the generation of terrain, 400 to the generation of cultural features on the terrain, and 300 for the generation of moving models. Level-of-detail control is utilized on non-terrain objects to reduce polygon throughput by displaying distant objects with fewer polygons than close objects.

Texturing is available to give crew members sufficient motion queues to accurately estimate vehicle velocity and to add realism to the simulation. The system allows any and all polygonal surfaces to be textured with a perspectively correct texture pattern. Up to 256 different two dimensional texture patterns can be stored in the system. Patterns can be generated from any two dimensional image, such as a photograph.

A transparency feature enhances the usefulness of texturing and allows such things as semi-transparent trees. To reduce the polygon throughput of the system, stamp textures are also permitted. A stamp is a single textured polygonal surface that rotates to face the viewpoint and is used to model single trees and various special effects.

Major cost reduction is achieved by minimizing the calculated image resolution and pixel repeating to fill the desired screen area. The seven visual channels with a 2.5 to 1 aspect ratio have calculated resolutions of 320 horizontal elements by 128 vertical elements. The gunner's channel is calculated at a 200 by 200 resolution. The images are anti-aliased in order to reduce the undesirable visual effects generated by the low resolutions and to allow the viewing of subpixel size objects.

In a simulated image a ray passed through a given screen pixel may pass through multiple objects. This phenomenon, known as depth complexity, is of prime concern in a ground based simulation. If an image is created from a high altitude, a typical worst case ray will pass through one object and the ground. In a ground based simulation a ray may pass through a tree, then a building, followed by a tank, and finally terrain. Simnet allows for an average depth complexity of 3.8 over the entire image. Sky areas of the image have depth complexities of zero as they are inserted after the image has been tiled.

The chosen viewing range in the visual system is 3500 meters. Distance fading is utilized so no distinct edge of the world is visible. The initial SIMNET database is a 50 km by 50 km area. The system dynamically updates the on line local area from disk so the crew can maneuver anywhere in the database. All geometric operations are performed in 32 bit floating point math to reduce positional error.

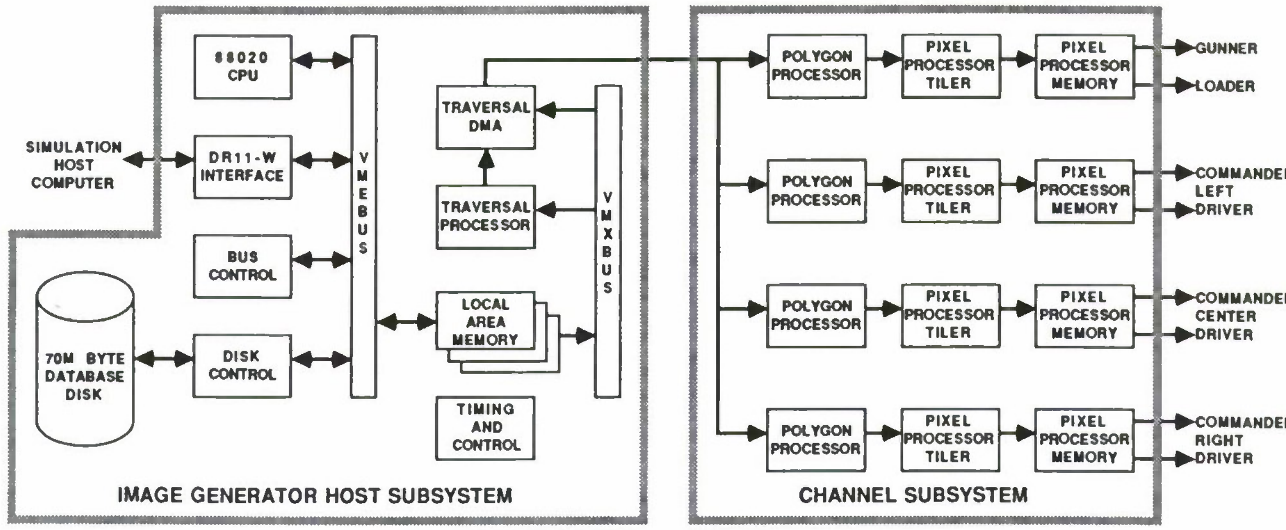
The system operates at a 15 Hz. image update rate. Transport delay through the system is less than 150 milliseconds from receipt of the data until the image is fully displayed on the screen.

Additional system features include the ability to send a description of the terrain surrounding the vehicle to the Simulation Host Computer, the calculation of ballistics intersections with all objects in the database, and the return of a laser range distance.

SYSTEM OVERVIEW

A block diagram of the SIMNET Visual System is shown in Figure 1. The system is composed of two basic units: the image Generator Host Subsystem and the Channel Subsystem. General tasks needed by all image channels, as well as communication with the simulation host computer, are performed in the Host Subsystem. Actual image generation for the eight visual channels is performed in the four parallel paths of the Channel Subsystem.

The Image Generator Host Subsystem is a VME/VMX bus based computation unit residing in a 19 inch, double height VME chassis. A Motorola 68020 CPU board, with 68881 floating point coprocessor and 1M byte of memory, resides on the VME bus. Also on the bus are a bus controller board, DR11-W parallel interface board, disk controller board connected to a 70M byte Winchester hard disk, and three dual ported memory boards. Each memory board contains 512K bytes of fast static RAM. The second port of the memory boards is connected to the VMX bus. Residing on the VMX bus with the memory boards is a two board set of custom hardware used to traverse the database.



SIMNET VISUAL SYSTEM BLOCK DIAGRAM
FIGURE 1

Communication with the Simulation Host Computer is performed over a DR11-W parallel interface. The visual system initiates a message transfer each frame by sending the current calculated laser range word, any ballistic intersection that occurred during the frame, and a local terrain message (if required) to the Simulation Host Computer. The local terrain message is a description of the terrain surrounding the simulated vehicle and is sent approximately once per second. It is used by the host to determine vehicle orientation, the mobility characteristics of the terrain, and vehicle collision with the environment. The host then sends a message to the visual system that includes its own vehicle position and orientation data, other moving models' position and orientation data, special effects positions, laser range display data, ballistic chords, and visual system configuration control commands.

Real-time software running in the 68020 CPU is responsible for controlling the visual system. In each frame it must build and transmit the outgoing message packet. Upon receiving the incoming message packet, it transfers data to the correct location in the double buffered section of local area memory to allow the display hardware to properly create an image. If a ballistic round fired from "my vehicle" is in flight, the CPU determines any intersection of the round's path with the environment. Finally, the CPU must determine if the simulated vehicle has moved sufficiently to necessitate retrieving new portions of the database from disk and storing them in Local Area Memory. Additional non real-time tasks running on the CPU include diagnostics, new database downloading, and system initialization.

Local Area Memory serves as the interface between the real-time software control functions and the display hardware. The 512K bytes of memory on each board are configured as 128K 32 bit words. The memory space is partitioned into a number of logical blocks. In the largest block 256K words are allocated to unique database storage. This includes a description of the terrain and cultural features within the 3.5 km viewing range of the vehicle. This block is updated as "my vehicle" moves through the environment. In a second block, 64K words are allocated to generic object storage. This block contains descriptions of all objects that may be displayed in multiple locations and is read from disk during system initialization. Examples of such objects are houses, power towers, and all moving models.

The remaining portion of memory is divided into two sections and is used as double buffered storage. During a frame, the CPU places data in one buffer while the display hardware reads from the second buffer. At the beginning of a frame the buffers are switched. Data placed in the buffer includes own vehicle position and orientation, moving model lists, and special effects tables.

The two board Traversal Processor and Traversal DMA card set is responsible for traversing the database and sending data to each of the four paths of the channel subsystem. In each frame the processor determines the fields-of-view of each of the eight channels. It then examines the database, including moving model and special effects tables, and performs field-of-view and level-of-detail tests on objects. When the data to be sent to a channel is determined, the processor sends a pointer to the DMA board indicating how

many words to send to a channel and the beginning memory address of the data. The DMA board stores these pointers in a series of pointer queues until the correct channel path can accept the data. It then retrieves the data from memory and sends it to the appropriate channel.

An additional card residing in the Image Generator Host Subsystem controls all synchronization and timing functions in the simulator; it interrupts the CPU to indicate the need for a new message transfer to the Simulation Host Computer, switches the double buffer portion of Local Area Memory, and resets the various boards of the display hardware path in such a manner that overall system throughput is maximized.

The Channel Subsystem resides in two 16 inch high custom chassis. Each chassis contains two independent display paths and each display path processes data for two preassigned channels. A display path is made up of three custom boards. These boards are the Polygon Processor, the Pixel Processor Tiler, and the Pixel Processor Memory.

A Polygon Processor board is made up of two pipelined microcoded engines. The board is responsible for transforming all terrain, dynamic and static model data, and special effects data to viewpoint space. Polygons are formed from the transformed data with backfacing polygons removed from further processing. The remaining polygons are then clipped to the pyramid of vision and are perspectivly projected onto screen coordinates. Calculations needed to prepare the polygons for the polygon fill operation and to add texture are performed. The polygons are then passed to the Tiler.

All calculations within the Polygon Processor are performed using IEEE standard 32 bit floating point math. Each board can perform at a rate of 40 MFLOPS and can output 30,000 front facing, clipped, four-sided, textured polygons per second.

The second board in the channel pipeline is the Pixel Processor Tiler. This board receives three or four-sided polygon descriptions from the Polygon Processor and places them in a polygon queue. The queue helps smooth the flow of data between the Polygon Processor, which processes data at a constant polygon rate, and the Tiler, whose polygon processing rate is determined by the number of pixels in the polygons. From the queue, a polygon is passed to the tiling engine where the screen pixels within the polygon are determined. For each pixel the Tiler determines a weight value, used to anti-alias edges, and a depth value, used for hidden surface elimination. Pixel color is set to a preassigned value or is looked up from a texture pattern table stored in EPROM on the board. Distance fading is applied prior to outputting each pixel.

The Tiler board consists primarily of MSI and LSI circuitry, memory devices, and 13 custom gate arrays of two different designs. Each Tiler has a throughput of 5.5 million pixels per second.

Pixel data consisting of a screen address, channel bit, color, weight value, depth, and priority are passed from the Tiler to the Pixel Processor Memory. This board, working at the same speed as the Tiler, performs hidden surface elimination, color averaging for antialiasing, and frame buffer storage. A hybrid depth buffer algorithm, utilizing the depth and priority information, is used to perform hidden surface elimination.

The board contains five separate 64K word memory banks. Two banks are used to double buffer the color and weight information for a channel. This double buffering allows an image to be built in one buffer while the image built during the previous frame is output to the display. Two additional banks are used for double buffering of the second channel. The last bank is used to store depth and priority data and is shared by the two channels. This sharing requires that the primary channel's image be completed before the secondary channel's image is processed.

Video circuitry produces RS170 standard R, G, B, and sync outputs for each channel. SIMNET images are produced with 320 by 128 calculated pixels (except for the gunner's channel which is 200 by 200) and at a frame rate of 15 Hz. The video circuitry displays the same data to each field of the RS170 output, and times the horizontal data such that the 320 pixels fill an entire line. This allows coverage of 640 by 256 RS170 pixel elements. The same data is displayed for two RS170 frames (RS170 frame rate is 30 Hz.) so no screen flicker is visible. It should be noted that the entire visual system can operate at 30 Hz. frame rate with approximately half the polygon throughput as in the SIMNET specification.

The entire system resides in a single, six foot high, 19 inch rack. In addition to the VME and two Channel Subsystem chassis, the rack contains a video distribution panel, power panel, and blower assembly. The system operates in a normal air conditioned environment using standard 120 or 240 volt power.

HIDDEN SURFACE ELIMINATION

Two SIMNET system requirements influenced the method of hidden surface elimination that was chosen for use in the SIMNET Visual System. The first is the large number of moving models that are free to move anywhere in the database. The second is the desire to rapidly build large databases which closely match actual terrain. To meet these requirements within the system cost guidelines, a hybrid depth buffer, hidden surface elimination algorithm is utilized.

Conventional real-time simulation devices use a priority sort algorithm in which surfaces are ordered, usually front to back, before polygons are filled to perform hidden surface elimination. By using this method, only the first pixel written to a screen location is saved. The pixels that arrive later are discarded or, in some systems, never passed to the frame buffer.

Advantages of this method include the simplicity of antialiasing, due to the coherence in data ordering, and the ability to suspend the polygon fill operation of hidden surfaces. Disadvantages include the difficulty of displaying moving models as well as the increased complexity of the database generation task. Moving models are more difficult to handle as each must be ordered with all other database objects. Database generation is more complex because all objects must be sortable and specific aids to help in the sorting of the polygons must be incorporated.

In a priority sort database no two polygons may intersect each other. Two intersecting polygons are not sortable and will cause errors in hidden surface elimination. With this constraint, it is impossible to do things such as burying the botton of a building in the terrain. Instead, the building edge must be contoured to the terrain on which it is placed. This is not a major restriction if the terrain is flat, but does become very diffcult if it is sloped. In addition, each building then becomes a unique object and will require additional database storage.

The depth buffer algorithm solves the hidden surface elimination probelm on a pixel by pixel basis. While tiling an image, the distance from the viewing plane to each pixel is computed. The pixels are compared in the depth buffer and only the closest pixel is retained for each screen element. Disadvantages of the depth buffer method are the fact that depth values must be calculated and stored for each pixel, and the difficulties that arise in antialiasing an image due to the loss of data ordering. Polygons do not have to be sorted, however, and intersecting surfaces can be processed just like all other surfaces without hidden surface problems.

The SIMNET Visual System uses a combination of depth and priority bits in its calculations. Each polygon is assigned one of eight priority levels. During the depth compare, if two pixels are within a given tolerance, the priority bits are used to determine visibility. Tolerances are adjusted depending on the priority levels of the stored and new pixels.

By using this method, surfaces can be coplaner and still yield the desired visual scene. Examples of this are roads overlayed on terrain and windows on buildings. In these cases, the road or window would be assigned a higher priority than the terrain or building. Without this ability, the road would have to float over the terrain by some finite distance. This could create the undesirable effect of being able to see under the road from certain viewpoints.

Incremental depth values are computed in inverse depth space inside the Tiler by using a 32 bit fixed point accumulator. In perspective space, depth is not linear, but inverse depth is. The values are inverted and a 16 bit depth value is passed to the Pixel Processor Memory. This provides a depth resolution of 1/16 of a meter with the SIMNET viewing range. It should be noted that the calculation and inversion of depth is also needed for proper perspective texturing.

THE DATABASE AND DATABASE GENERATION

A major goal of the SIMNET design was the ability to rapidly generate databases that could be efficiently displayed on the visual simulator. This rapid development allows large databases to be built for many different locations. If the databases are built to match real world locations, true mission rehearsal is possible.

To achieve this it was necessary to develop a set of user friendly tools to assist the database modeler and to automate as many tasks as possible. BBN Delta Graphics has developed a set of database development tools that help meet this goal.

As stated earlier, terrain is generated from DTED data. A DTED tape is read and automatically resampled to the desired grid spacing. In SIMNET the spacing is 125 meters. Terrain relaxation is performed on the resampled data using tolerance parameters selected by the modeler. Terrain is then represented by a grid of elevation points and a list of three and four-sided polygons which interconnnect these points. Information needed to place the desired texture pattern on terrain is then added to each polygon.

Three dimensional models are developed with a CAD package optimized to build simulation models. Data describing the models can be entered as numerical components, via a mouse, or by using two or three dimensional digitizing devices. Both RGB and HSL color tools are provided to assist in the selection of model color. Alternately, texture patterns can be placed on any surface.

Database assembly is performed by first calling up the desired piece of terrain. Objects can be placed on the terrain by using numerical components, a mouse, or a digitizing table. The modeler picks the X and Y coordinates on the terrain where the model is to appear and also selects the orientation. The software automatically determines the Z coordinate.

The random placement of objects is performed by selecting an area and density for the objects and letting the software determine exact placement. This works quite well for trees and other natural objects whose exact location is not important. Roads and river networks are entered as a series of lines on the two dimensional terrain plane. The modeler selects width and texture, or color, for the network. The software then expands the road or river network to the desired width, contours it to fit exactly over the terrain, builds road intersections, and applies the texture or color data.

All of the above operations are performed with data in a very generic format. The same formats are used to create databases for visual, IR, and radar simulators as well as commercial animation sequences. To prepare the data for display on the SIMNET Visual System, it must be processed by the appropriate database compiler. The compiler organizes the data so it can be efficiently traversed and displayed.

The final SIMNET runtime database is composed of .5 km by .5 km patches of terrain, all unique models, and the pointers to all generic models that reside on it. Each patch is referred to as a load module. Data contained in a load module is composed of two types: traversal commands and geometric data.

The Traversal Processor looks at traversal commands, but never at the geometric data. Typical traversal commands tell the Traversal Processor to jump to a new database address, perform a field-of-view or level-of-detail test and jump to a certain address if it passes, jump to generic memory, or output a specific number of words to a Polygon Processor. The Polygon Processor sees only geometric data. This data consists of matrix manipulation commands and data describing terrain, polygon, and stamp components.

DATABASE TRAVERSAL

During each frame a decision must be made as to what data to send to each of the eight visual channels for processing. This data must include moving and static models, terrain, and special effects. While ordering of the data is not required by the hidden surface elimination algorithm, it is desirable for other reasons.

First, as will be described in a later section, the method of antialiasing used in the SIMNET System performs better if data is ordered front to back. Second, front to back ordering is preferred in the event of overload. If an overload condition develops, the system will stop processing data in order to begin the next frame. If this occurs, data processed at the end of a frame time may be lost. If processing is front to back, the data lost will be that farthest from the viewpoint.

Load modules are stored in Local Area Memory in a 16 by 16 element array with each load module being allocated 4K bytes of memory. Because a load module is always placed in the same memory address, all addressing within the load module is absolute and is assigned at the time of database compilation.

The load module that "my vehicle" is on is referred to as the base load module. Active load modules in the array are all those on the same row or column as the base load module and those within seven load modules in all directions. Thus, a 15 by 15 array is active and will contain data for all terrain and objects within 3.5 km of "my vehicle". Data for the remaining row and column is being retrieved from disk, as part of the dynamic database update process.

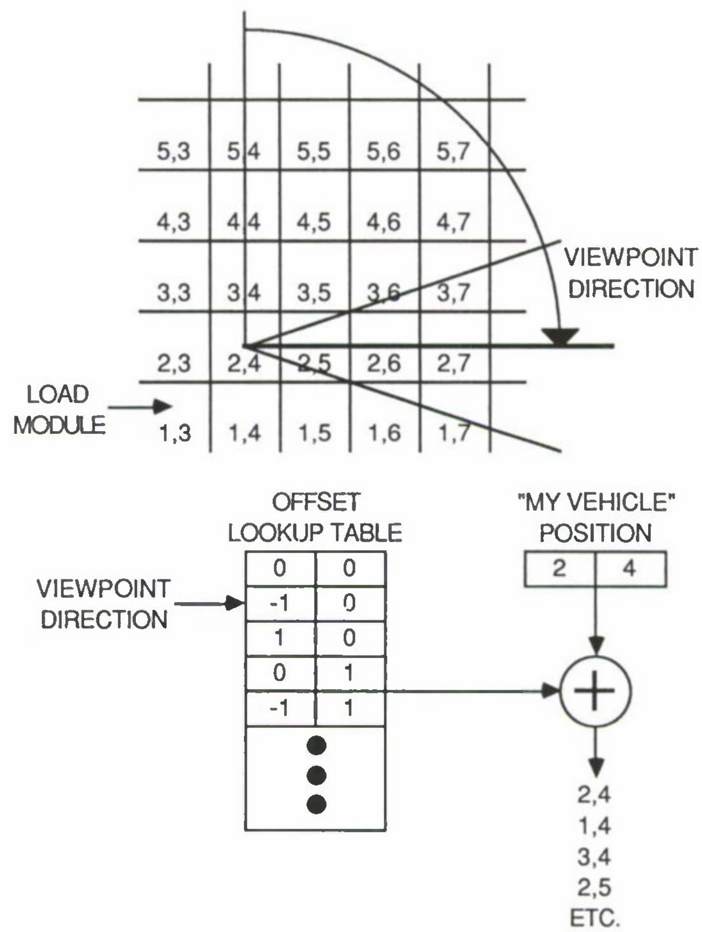
When moving models and special effects are placed in Local Area Memory, the real-time software determines which load module they reside on. It then builds a series of moving model tables; one for each load module. The software also places "my vehicle" information in memory.

At the start of a frame the Traversal processor determines field-of-view constants and viewpoint direction data for each channel. Load modules for each channel are then processed front to back by using a lookup table. This table indicates the next, potentially visible, load module offset based on the viewpoint direction. This offset is added to the address of the base load module and creates a pointer to the next load module to be processed. See Figure 2.

A field-of-view test is then performed on the entire load module. If the test passes, the moving model list for that load module and the load module are processed. To assure even distribution of the data to the Polygon Processors, the Traversal Processor interleaves the processing of load modules for the various channels.

ANTIALIASING

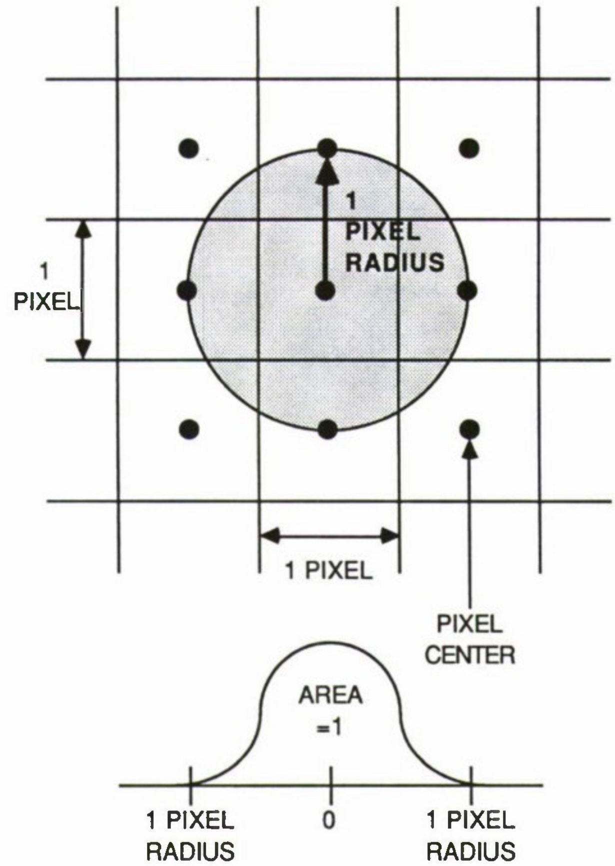
Antialiasing is the process of removing the disturbing visual effects that result from discreetly sampling an image. These effects would be quite noticeable in the SIMNET System because of the low screen resolution employed. Traditional methods of antialiasing a depth buffer system require that pixels be supersampled, usually by a factor of 16 to 1, and that a depth buffer location be used to store each subpixel value. This method, although effective, does not meet the cost requirements of the SIMNET System. Therefore, a method of weighted oversampling and filtering is used.



DATABASE TRAVERSAL
FIGURE 2

During the tiling process, a weight value is calculated for each pixel within a one pixel radius of the polygon being tiled. The weight value is determined by calculating the filtered area of the pixel that is covered by the polygon. The filter function is stored in lookup tables on the Tiler board and has a radius of one pixel. See Figure 3. The weight value is then sent to the Pixel Processor Memory board along with the other pixel information. Here, the depth and weight values of the new and stored pixels are used to determine the correct blend of pixel colors for producing a final color. The weight value is also used in the processing of transparencies.

This method works well when pixels arriving at the memory are ordered front to back. If ordering is not present, it is possible to see distant objects bleeding through the polygon boundaries of closer objects. The front to back load module processing, as well as the ordering of objects on a load module, helps reduce this problem. For example, on a load module the three dimensional models (buildings, etc.) are processed first. Road and river networks are processed next, followed by terrain. An additonal feature, which allows erasing of the screen in a local area before an object is processed, also reduces the bleed through problem.



ANTIALIASING FILTER
FIGURE 3

ENHANCEMENTS

The ultimate SIMNET battlefield will contain not only M-1 tanks as well as other ground based vehicles and dismounted infantry, but also fixed and rotary winged aircraft. The goal is to allow training under all potential battlefield environments. These include full day, dusk, and night simulation with adverse weather and manmade environmental effects such as blowing and expanding smoke clouds used for cover and concealment. Programs to improve the performance of the SIMNET Visual System to meet these goals and to allow the system to be used to simulate vehicles other than the M-1 have been completed or are underway.

Making the system more flexible and easier to upgrade was a major desire. In the initial system, microcode, and most look up tables, including texture storage, were stored in permanent memory devices. If changes were desired, new PROMs had to be made and placed in systems. Enhancements will place microcode and look up table data in RAM.

During system initialization, this data will be read from disk and passed to the appropriate hardware. Field upgrades can then be accomplished by sending the new data over the SIMNET network. All simulators will receive the data and copy it to their disks at the same time. This will greatly simplify the upgrade task. This also allows different databases to use different texture patterns as they are downloaded.

Downloadable microcode along with improvements in the Traversal Processor algorithm make it simple to reconfigure the visual system to simulate other vehicles. To date, the system has been used to simulate the M-2 and M-3 Bradley Fighting Vehicles, a generic helicopter, and the M-1.

To allow higher resolution simulation a 640 by 480 pixel system has been developed. This system, the Delta 120TX/T, uses the basic SIMNET design. The four Pixel Processor Memory boards are replaced by two new memory boards and a video and 2D overlay board. The system provides a single channel of imagery at 30 Hz. with 3000 potentially visible polygons per frame. In addition, a separate 2D path exists to place overlays on the screen without degrading the system's 3D performance.

Other system enhancements will include increases in the size of local area memory, improved stability in textured data, IR simulation, a fifty percent increase in polygon throughput, and more powerful simulation host processing.

CONCLUSION

The SIMNET Visual System described in this paper was designed over a 15 month period from late 1984 through 1985. As of June 1987 over 75 systems have been manufactured and delivered to sites in Fort Knox, Kentucky and Grafenwohr, West Germany.

The use of selective fidelity and the accompanying tradeoffs in visual system performance have proven to be an extremely effective method for designing low cost training systems. At the same time, ongoing programs to enhance the visual system hardware and database construction software promise significant improvements in system performance and database construction speeds.

The systems have received outstanding acceptance from the crews actively involved in training. Said Pfc. Bill Manuel, a SIMNET trainer, "I came out here to look at the range this week and I was surprised. They had 301 (the range) almost identical on SIMNET, except it was animation."

Furthermore, in June of 1987, teams for the United States Army placed first and third in the Canadian Army Trophy Competition (a biannual tank contest held in West Germany) after training extensively on SIMNET systems. Sample

images taken from the SIMNET Visual System are shown in Figure 4.

ABOUT THE AUTHOR

Richard Johnston is Systems Engineering Manager for BBN Delta Graphics Inc. He was responsible for the architecture of the SIMNET Visual System and is currently involved in architectural studies of future systems. Mr. Johnston holds a Masters of Science degree in Electrical Engineering from Georgia Tech and has a background in VLSI circuit design, digital signal processing, microcoding, and Computer Image Generation. Before joining Delta Graphics, Mr. Johnston worked for the Boeing Aerospace Company where he was responsible for the architecture of the B-1 flight simulator's visual system.

SIMNET IMAGES
FIGURE 4

PHOTOGRAPHIC TEXTURE AND CIG: MODELING STRATEGIES FOR PRODUCTION DATA BASES

L. Charles Clark
Thomas C. Brown

Evans and Sutherland Computer Corporation
Simulation Division
Salt Lake City, Utah

ABSTRACT

The addition of texture to Computer Image Generation (CIG) systems has increased the potential for realism and cuing effectiveness in visual data bases used for flight simulation. While the visual simulation industry has already embraced texture technology, most of its attention has been focused on synthetic or statistical patterns. The use of photographic texture has been demonstrated and shows great promise, but it has not yet been thoroughly exploited in the production environment. Although photographic texture can significantly enhance the realism of a data base, its indiscriminate use often introduces unrealistic visual anomalies into the scene. However, when it is applied correctly, photographic texture can improve the efficacy of current and future CIG systems. The enhanced realism in flight simulation which accrues from the proper use of photographic texture provides a critical advantage in training effectiveness.

This paper discusses the scope of usefulness for photographic texture in production data bases, particularly for constructing self-repeating texture patterns. The results of new modeling strategies which mitigate or eliminate some of the visual anomalies inherent in the use of photographic texture are also described. Finally, examples are given of how photographic texture can be exploited to meet some specific training requirements for current and future flight simulators.

INTRODUCTION

The concept of texture, as discussed in this paper, refers to a way of modifying the surface characteristics of a polygon based on information stored in a two-dimensional array of data called a *texture map*. Each texture element in the two-dimensional array is called a *texel*. Generally, more than one texture map can be used to modify a polygon. In the Evans and Sutherland CT6 image generator, a set of maps applied to a single polygon is called a *mapset*, and may include up to four maps. By varying the scales and positions of the maps in a mapset, a great variety of patterns can be achieved with only a few maps.

In CT6 systems, texture is available in two forms: *modulation texture* and *contour texture*. Modulation texture varies a polygon's attributes almost continuously from one state or region to another, while contour texture is used to define a distinct edge between one state and another (like a cookie cutter). A combination of these two types of texture on a single polygon can create a very realistic image. An example is a tree that has subtly varying foliage colors while maintaining crisp branch and leaf shapes regardless of the proximity of the eye to the tree.

A polygon has three basic surface characteristics that can be modified with texture: intensity, color, and transparency. The standard application of texture is to vary intensity. In this case, a texture map varies the brightness across a polygon, texel by texel, based on the corresponding intensities stored in the map. Color can also be varied smoothly across the surface of a polygon, blending from one color to another. Finally, texture can be used to vary the transparency of a polygon. Both contour and modulation texture may vary each of these three characteristics, and both types may be combined in a mapset to provide a wide variety of visual results. (7) In this paper we will focus on the construction and use of texture maps for terrain color modulation, i.e., texture which modulates between colors on terrain surfaces.

Approaches to Making Texture Maps

Photodigitizing is one of two general approaches to making texture maps. The other approach, *synthetic generation*, is well understood in the visual simulation industry, and synthetically generated texture maps have been applied successfully in production data bases. Some approaches to generating synthetic texture include hand digitizing, procedural methods for creating simple patterns, statistical methods for generating patterns which imitate nature in the frequency domain, and fractals for imitating nature in the spatial domain (4).

Photodigitizing, on the other hand, is not as well understood nor has it been used as extensively, except perhaps to produce marketing material. The extent of the difficulties inherent with the use of photo-derived texture depends on what class of texture pattern is being created.

Classes of Texture Patterns

There are two general classes of texture patterns: *discrete* and *self-repeating*. Discrete patterns are used for modeling single occurrences of distinct objects (e.g., trees, people, aircraft insignia, etc.). Such patterns are typically easy to construct with either synthetic or photodigitizing procedures, and applying them in a visual data base is very

straightforward. Although photodigitized discrete patterns are simple to construct and apply, they are not without problems. However, these problems, (which include obtaining orthographic source photos, and editing their background, color, and content registration) are easily managed in a "piece work" fashion.

Since the on-line memory for storing texture maps in a CIG system is always limited (5), it is virtually impossible to texture an entire production-size data base using discrete texture patterns. Some textured surfaces cover very large areas -- for example, terrain, clouds, and water. If such areas were textured with discrete patterns, an exorbitant amount of on-line memory would be required to store them. As a result, large surface areas are usually textured with self-repeating patterns. However, these self-repeating patterns are almost always synthetically generated, not photodigitized. The photo-derived texture often demonstrated in marketing material usually consists of local splashes of discrete patterns, possibly iterated over and over throughout a data base. We have found that this use of texture falls short of the real potential for using photo-derived texture in production data bases.

Unfortunately, photo-derived texture patterns which self-repeat are much more difficult to construct than are discrete patterns. In a self-repeating texture map, the edges must be blended to match left to right and top to bottom. This makes the boundaries between replications of the pattern inconspicuous. In addition, the contents of the map must be kept as homogeneous as possible to prevent the replication of some dominant feature in the texture map from emphasizing the inherent periodicity of the self-repeating pattern. (6)

The Challenge: Creating Photo-Derived Self-Repeating Patterns

Self-repeating texture can be readily constructed using the synthetic methods mentioned previously which allow the modeler to control the blending of edges and the homogeneity of content as the map is generated. However, in a photograph, the pre-existing edge and content conditions are usually far from optimal for natural self-repetition. One seemingly obvious solution to this problem would be to "set up" the source photos so that they match. However, this process would be expensive and complex.

Another possible approach would be to alter the content of the photos in the darkroom or with an airbrush so that the edges match. Techniques for manually altering photos have been around for as long as photography, and have been improved by modern electronic technology. However, manual editing would be tedious, time consuming, and extremely expensive, especially for a large number of photos.

The most rational approach to generating self-repeating patterns from photos is undoubtedly to use some digital method to blend the opposite edges of an image so they match in a natural looking way. The evolution of possible approaches to this edge-blending problem is discussed next.

FINDING SOLUTIONS: AN OVERVIEW OF EXISTING METHODS

Although the addition of texture to CIG systems has just recently brought the edge-blending problem into focus for the simulation industry, much thoughtful progress toward its solution has already been achieved in the field of image processing. For example, photomosaics have been constructed with multiple overlapped images from interplanetary space probes, earth satellites, and telescope photography. (1) Existing techniques for constructing photomosaics are the basis for solving the edge-blending problem.

The Multiresolution Spline Approach

A technical problem common to the construction of all photomosaics is joining together two images so that the edge between them is not visible. This problem can be solved with an image spline. An image spline is any digital technique for gently distorting the edges of adjacent and usually overlapping images so that they may be joined together with a smooth seam. Probably the single most important contribution to image spline technology is the recognition that edge blending is best done as a separate process for each frequency band in an image. This approach, called the multiresolution spline, was conceived and developed by Burt and Adelson (1). In the multiresolution spline, each image is first decomposed into a set of band-pass frequency images called a Laplacian pyramid. Then the Laplacian pyramids for the images are spliced together into a new Laplacian pyramid where texel values are averaged along the splice boundary. Finally, the new Laplacian pyramid is recomposed, yielding a successfully splined image.

Burt and Adelson showed that the multi-resolution spline can be used to successfully join the following types of images.

- *Similar overlapped images*, e.g., two Landsat images with identical content but different gray levels.
- *Dissimilar overlapped images*, e.g., pictures of an apple and an orange. Although these images are dissimilar in the frequency domain, they are spatially symmetrical and therefore lend themselves to being splined.
- *Similar non-overlapped images*, e.g., images composed of pixel-block subsets resulting from data compression. Although these pixel block subsets are non-overlapped, they do originate from a single image. This results in similar spatial frequencies and content along the shared boundaries of adjacent pixel blocks, which makes them easier to spline than a set of unrelated images.

Although the multiresolution spline successfully blended these categories of images, there is still one case unaddressed by Burt and Adelson -- the need to create a self-repeating texture pattern from a non-repeating photograph. This problem falls into the category of *dissimilar non-overlapped images*. We tried using the multiresolution spline for this case, and we found that a single image could not be

splined with itself using this method without noticeable discontinuities appearing between repetitions.

Yang et al.(6) also concluded that the use of the multiresolution spline to blend the opposite edges of a single image did not achieve acceptable results. They found that in the absence of image overlap, the extrapolation that occurs in the process of decomposing and recomposing the Laplacian pyramid placed too much weight on edge and corner texels and the edges did not blend inconspicuously. When the original image was overlapped with itself by 50%, blending along edges was somewhat improved. However, this approach resulted in changes to the general image character. In addition, it only worked for carefully selected images and was not considered to be a general solution.

The Multiresolution Image Pyramid

Another approach to solving the edge-blending problem was suggested by Zimmerman (7). He found that the multiresolution spline could be used to create a self-repeating pattern from a non-repeating image if an edge-blending filter was integrated into the process. In his implementation of the multiresolution spline, called the Multiresolution Image Pyramid (or MRIP), Zimmerman applied an edge-blending filter to the perimeter texels in each frequency band before the image was recombined.

Although we observed that MRIP was more successful at splining dissimilar non-overlapped images than any previous approach, we found that the edge filter dampened the image intensity range along the boundaries. This made the image appear blurry and gray along its edges. The problem of splining dissimilar non-overlapped images still remained.

A SOLUTION: MRIP WITH SOME NEW TWISTS

To generate self-repeating patterns from non-repeating photographs, we borrowed the basic ideas of the MRIP process, and added some new twists. The two most difficult technical challenges in our implementation included: 1) the design of a successful edge-blending filter and 2) the correct integration of the filter algorithm into the MRIP process.

After experimenting with a variety of edge-blending filters, we developed an intelligent filter which yields acceptable results. When this filter is applied along the edges of the band-pass image, a self-repeating pattern is generated from a non-repeating image by blending the top and bottom, and left and right, sides. This filter gently distorts opposite edges toward similarity while minimizing blurring or other changes to the fundamental character of the image. We were able to optimize the success of this filter by adjusting critical functions in the MRIP algorithm.

The new edge-blending filter, combined with the enhanced implementation of the MRIP algorithm, provides exciting results. Figures 1, 2, and 3 compare examples of non-repeating images of desert terrain scanned directly from an aerial photo to examples of self-repeating texture patterns generated from these images using our implementation of MRIP. These photographs illustrate how the intelligent edge-blending filter gently coerces a non-repeating pattern to be self-repeating while preserving the original image character along the newly filtered edges. The figures also show how the filter allows important features which are inevitably truncated at an image edge to grow or extend naturally into the opposite edge region.

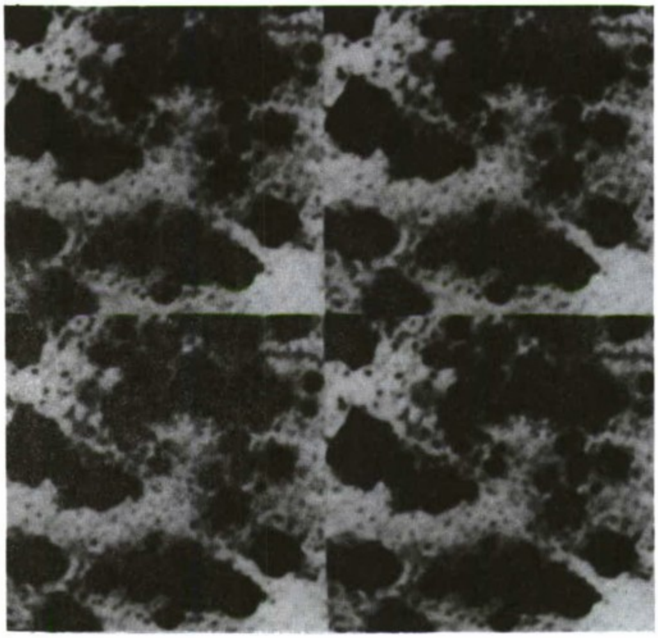


Figure 1a

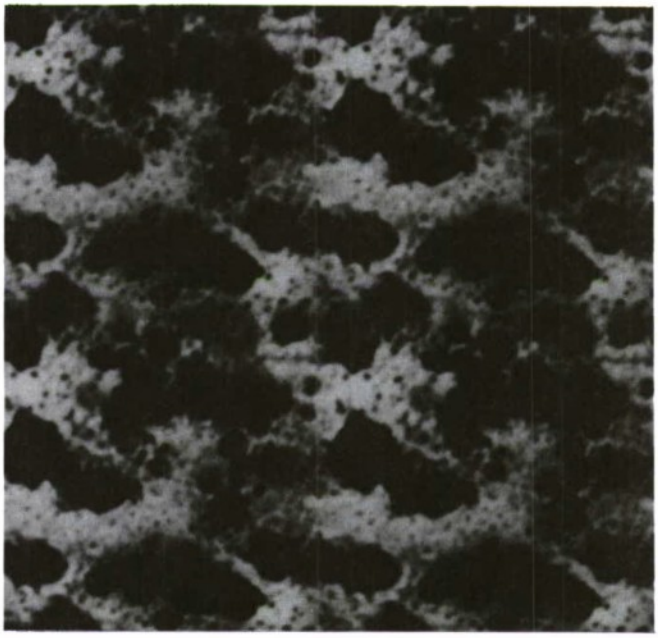


Figure 1b

Figures 1a and 1b show an aerial view of a small area covered with piñon/juniper trees. Dark objects in the photo are the tops of the trees. Figure 1a contains four copies of the photo and shows the problem of edge truncation in photo texture. Figure 1b is the same photo after the MRIP procedure has blended the edges to self-repeat. Note the truncated edge objects in Figure 1a and the subtle changes imposed in Figure 1b to facilitate edge blending.

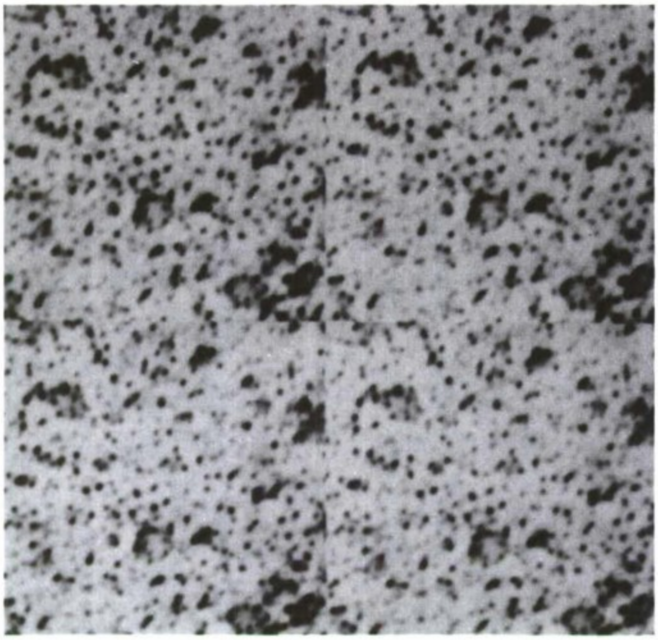


Figure 2 a

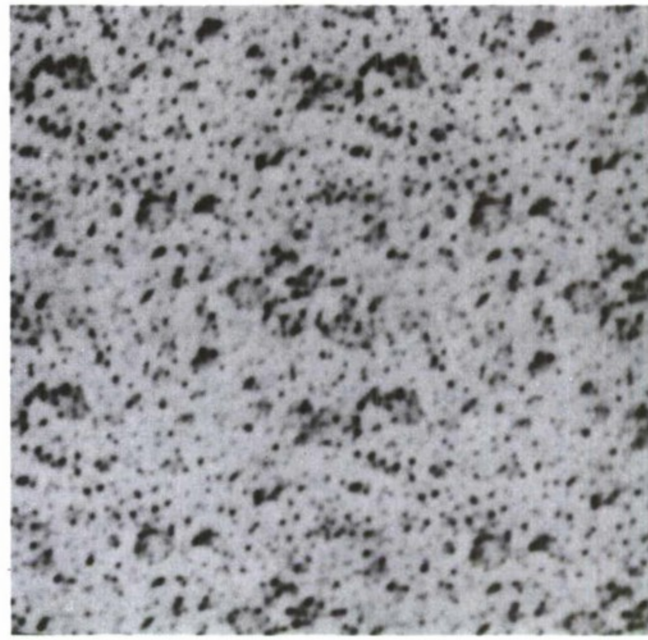


Figure 2b

Figures 2a and 2b show an aerial view of desert sagebrush. The sagebrush has a higher spatial frequency content than does the piñon map. Notice that in Figure 2a the higher frequency results in an original photo with nearly matching edges. Figure 2b is the sagebrush map, with fully matching edges, after the MRIP process is applied.

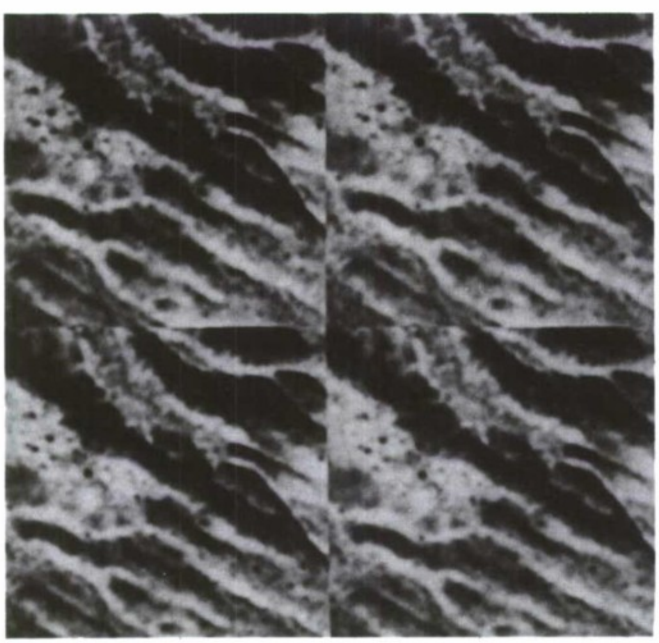


Figure 3a

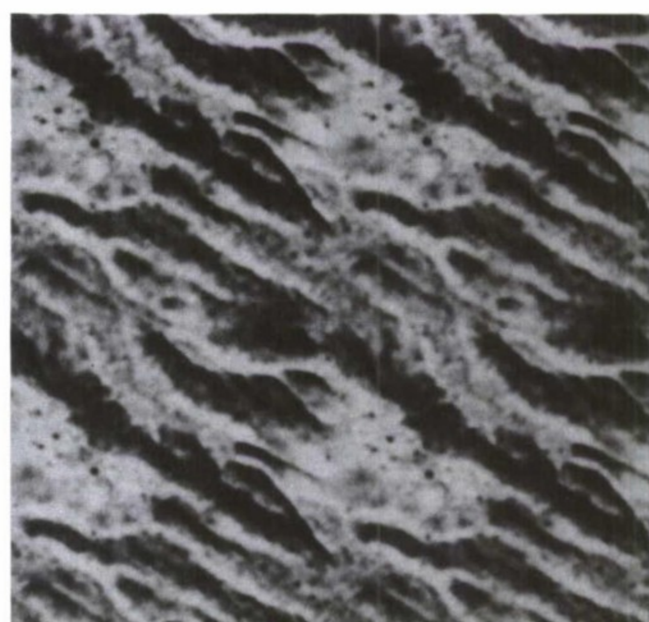


Figure 3b

Figures 3a and 3b are taken from an aerial view of desert relief. This view covers a much larger area than the view used for Figures 1 or 2. This arroyo pattern is applied at the largest scale of the mapset to give the lowest level of desert detail a greater illusion of topographic relief.

In addition to desert, we have found the MRIP procedure to be very successful at blending many other kinds of modulation and contour patterns. For example, Figures 4 and 5 illustrate the MRIP procedure applied to create photo-derived texture patterns of ocean surfaces.

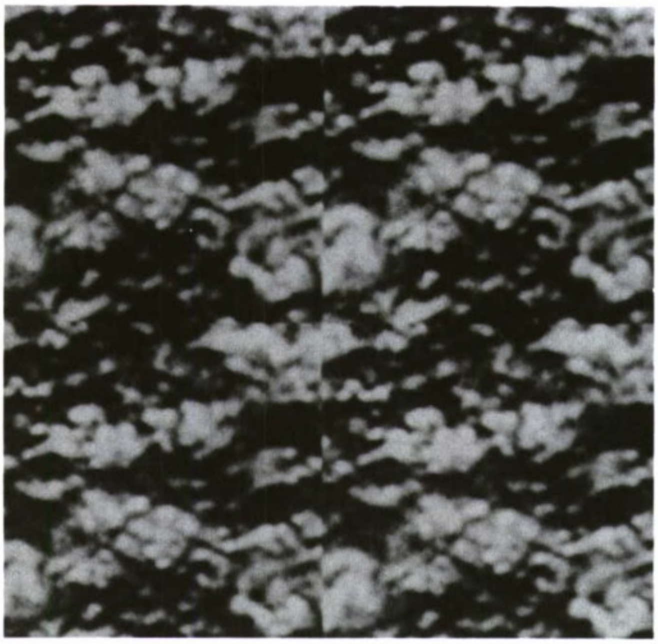


Figure 4a

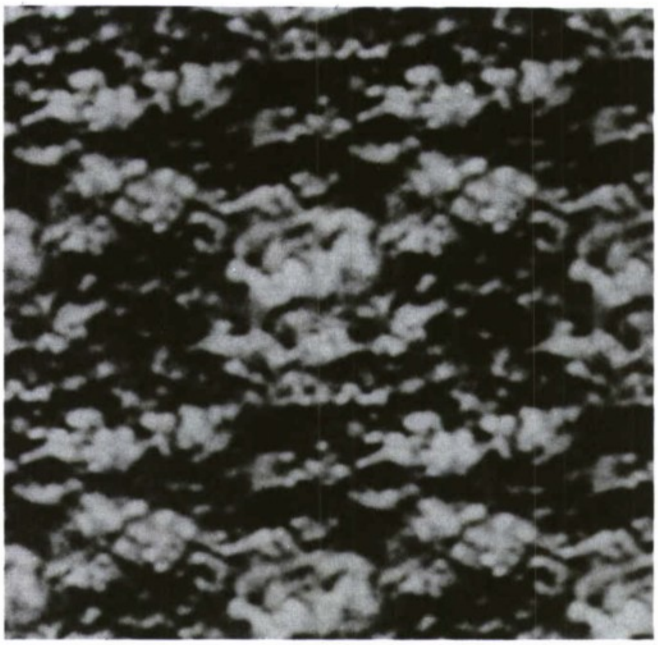


Figure 4b

Figure 5a

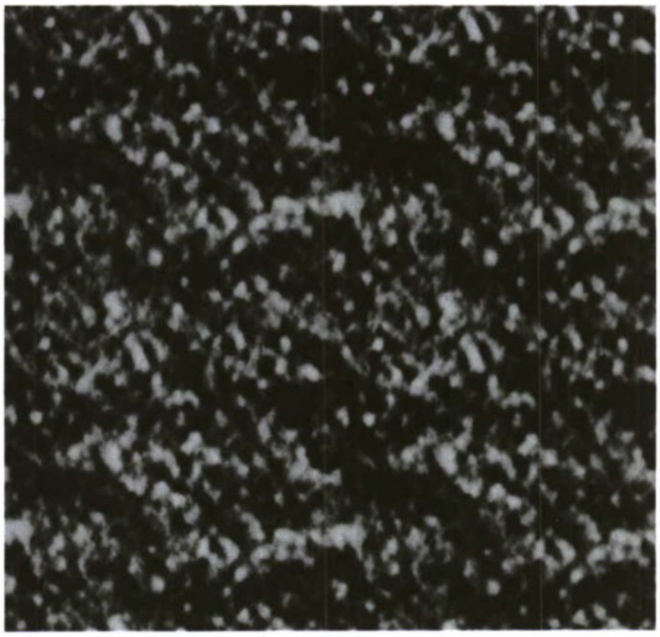


Figure 5b

Figures 4 and 5 are two photo textures of ocean taken from different altitudes, and consisting of slightly different sea states. Figure 4a is a photo of the ocean from a lower altitude and Figure 4b is the resulting texture map after the MRIP process. Figure 5a is a photo from a higher altitude with a calmer sea state and 5b is the resulting texture map.

IMPLEMENTATION STRATEGIES: OPTIMIZING VISUAL IMPACT

We have shown that the MRIP technique is very useful for creating self-repeating texture patterns from non-repeating photographic images. However, to insure that self-repeating texture maps yield optimal visual results, the following processes which relate to the construction and use of self-repeating patterns must be handled correctly.

- Acquiring the correct picture
- Selecting the correct image patch
- Determining texture levels of scale
- Applying self-repeating patterns in data bases

Acquiring the Correct Picture

The first problem in creating photo-derived texture maps from a photodigitized source is to acquire the "correct" (used as a relative term) photograph. Pictures for discrete patterns are easy to find, or just as easy to take. However, photographs which are suitable as image sources for self-repeating terrain texture patterns are not as trivial to acquire.

The most straightforward picture to use for generating terrain texture is usually taken from a high altitude with an orthographic view and precise control of scale. It is impractical for most of us to go take this kind of a picture. Fortunately, a large variety of scales and series of orthographic aerial photographs covering all 50 states is available from the Agricultural Stabilization and Conservation Service of the United States Department of Agriculture. These pictures contain images of topography, vegetation, water surfaces, urban areas, etc. They are invaluable image sources for generating terrain texture patterns.

Selecting The Correct Image Patch

Once the correct picture is acquired, the next step is to select the correct image patch. Rarely will an entire photograph be useful as a texture map even if it is the correct picture. More often, the best source for a texture map is some image patch within the picture.

For discrete patterns, the selection step is easy. The image patch can simply be selected, framed, and digitized. However, in the case of self-repeating patterns, image-patch selection is much more difficult and is critical to maximizing success. Selecting the correct image patch for use in terrain texture modeling requires the identification of patches which show the greatest potential for self-replication.

Two general approaches to selecting correct image patches include *statistical methods* and *inspection*. Statistical methods are particularly useful for identifying dominant features in an image patch which could potentially aggravate the periodicity problem. Such methods work on the assumption that an image patch which possesses strong potential for self-repetition exhibits the following characteristics: (6)

- The image patch has relatively constant correlational statistics over its entire surface.
- No subset of the patch can be easily and unambiguously segmented from the rest of the image.

These criteria generally limit image patches to isotropic content, i.e., similar frequency and spatial content in all directions. As a result, the texture patterns surviving these tests tend to be boring and bland. However, texture in the real world is often anisotropic, having form and content which result from and imply directionality. For example, arroyos in the desert, waves in the ocean, and cloud formations in the sky are evidence of anisotrophism in nature.

With anisotropic texture, the challenge in creating a self-repeating pattern is to align directional texture features at opposite map boundaries. This is best accomplished by inspection. In general, selecting the correct image patch is more a matter of art than science, and the best image patch selection tool is the human eye, along with a process of trial and error.

Determining Texture Level Of Scale.

A proven approach to masking the periodicity inherent in a self-repeating texture pattern is to combine two different scales of the pattern on a polygon. With this technique, each replication of the smaller pattern is subjected to a slightly different modulation of the large scale map than every other replication. (6) When four scales of the same pattern are combined together in a CT6 mapset, the resulting texture pattern begins to exhibit fractal-like behavior.

Fractals rely on the notion of self-similarity to produce models that mimic the natural world. The general concept of fractals is that most natural forms, such as a shoreline or the outline of a cloud, exhibit detail no matter how close the observer, and the form of that detail is similar regardless of scale. (7)

Although a fractal-like mapset (with four scales of the same texture pattern) on terrain polygons provides flight cues through a broad range of altitudes, it does not result in a truly realistic appearance. This is because fractals merely emulate a model of nature, rather than reproduce it, and there are too many natural phenomena to be defined in a set of fractal rules (4). For example, the macro-topography of an entire mountain range may be very similar in form to the micro-topography of a small foothill. However, in a textured terrain model, it is more important to show topographic texture patterns at large scales (or low frequencies), while it is more important to show vegetation patterns at small scales (or high frequencies).

Although fractals offer an interesting and useful approach to analyzing texture, we have found that the designation of texture map content for each level of scale in a mapset is best done through careful observation of nature, and common sense. In addition, the CT6 capability to combine four texture levels of scale in a single mapset yields extremely realistic texture effects.

Applying Self-Repeating Patterns in Data Bases

Given a mapset of self-repeating terrain texture patterns with appropriate content and level of scale, CT6 systems enable some very useful techniques for applying texture in a visual data base. These capabilities include the use of *macro texture*, and the unique ability of the CT6 to apply *smooth shading* to textured polygons.

Macro Texture. A common terrain modeling approach for CT6 data bases is the subdivision of terrain into a regular grid. The elevation data for this grid is typically derived from DMA terrain elevation data. (4) If texture is applied such that mapset repetitions begin and end at polygon boundaries, the repetition of the mapset and the faceting of the terrain model mutually accentuate each other. This undesirable interaction highlights rather than disguises the polygon boundaries in the terrain model. To remedy this problem, CT6 systems use macro texture.

Macro texture is the use of a mapset repetition interval which is larger than that of the polygonal subdivision of the terrain. In macro texture, the mapset is projected onto a multi-polygon terrain surface such that texture detail is continuous and unique across polygon boundaries. As a result, the largest scale pattern in the mapset covers many polygons. Up to three other texture maps at smaller scales may be overlaid onto this pattern to produce a broad range of texture scales. This produces a homogeneous texture pattern across the terrain which effectively masks texture map repetition and polygon boundaries.

Smooth Shading. The Evans and Sutherland CT6 is unique in its ability to render smooth-shaded textured polygons. When a mapset of photo-derived self-repeating texture patterns is applied as macro texture to a smooth-shaded terrain model, the illusion is complete. Polygon boundaries in the terrain model become virtually invisible. The result is a quantum leap in image realism, as shown in Figure 6.

OTHER APPLICATIONS FOR PHOTO-DERIVED TEXTURE

Most examples and explanations given in this paper have illustrated the use of photo-derived texture patterns for modeling desert terrain. Beyond the modeling of desert, an immediate potential exists for vastly increasing the realism of a variety of other terrain types modeled in visual data bases. The use of self-repeating photo-derived topography and vegetation patterns for models of forest, agricultural, and urban areas offers great promise.

In addition to terrain, other large surface areas in visual data bases which may be textured with self-repeating patterns include ocean and clouds. Figures 4 and 5 showed how the MRIP procedure could be successfully applied to ocean patterns. In addition to using MRIP, CT6 offers two capabilities that allow an ocean model to simulate sea states. (The term *sea states* refers to the frequency and magnitude of ocean waves.)

- Given a number of photo-derived texture maps representing a variety of sea states, the simulated sea state in a model can be changed by reloading texture maps.
- In addition, texture motion can be used to animate whatever texture maps are applied. (4)

Figure 7 shows the result of modeling an ocean surface using these capabilities with the texture patterns shown in Figures 4 and 5.

Great long-range potential exists for modeling terrain with photo-derived *discrete* patterns because of anticipated capabilities of future CIG systems. For example, CIG texture memory capacity will continue to increase. This capability, combined with dynamic updating of texture memory, will virtually eliminate texture memory limitations. This and other features will make it feasible to apply a mosaic of discrete map-correlating texture patterns to large terrain regions in a visual data base. The MRIP technology will play an important part in this capability, accomplishing the task of blending the edges of a large number of adjacent images.

Applications will be found for both contour and modulation maps in terrain image mosaics. Modulation maps will continue to be exploited for representing topography and vegetation patterns. Contour maps will prove invaluable for modeling sharp-edged features such as coastline. Both kinds of maps will be generated from a variety of sources, i.e., they may be photo-derived, or they may instead be generated from photo-like sources such as remotely sensed digital imagery (2) or DMA data.

CONCLUSION

The addition of texture capabilities to CIG systems has made possible the application of texture patterns to large terrain surface areas in visual data bases. However, the relatively limited texture map memory in today's CIG systems does not allow these large terrain regions to be textured with discrete (non-repeating) patterns, so self-repeating patterns must be used instead. Because the creation of self-repeating texture patterns from non-repeating photographic images has offered long-standing technical challenges, photo-derived texture patterns have not been as generally applied in visual data bases as have synthetically generated patterns.

A method for generating self-repeating texture patterns from non-repeating photographic images has finally been developed. This process, known as MRIP, is based on the concept that blending opposite image edges, as is required in generating a self-repeating pattern, is best done independently for each frequency band in an image. The key to the success of the MRIP approach is the use of an intelligent edge-blending filter which gently distorts the opposite edges of an image toward similarity while preserving the original image character along the newly filtered edges.

Given the capability to create self-repeating photo-derived texture maps, several implementation strategies insure their successful application in terrain data bases. The processes of selecting an

Figure 6

Figure 6 was photographed directly from the display of a CT6 real-time image generator. This view of the data base includes only 42 visible ground polygons, and is part of a completed production data base which contains over 100,000 polygons in its high-level-of-detail terrain model. Notice the piñon/juniper texture in the foreground and the arroyo map in the distant background. The 3D textured trees are added to allow the eye to calibrate to the correct scale of the piñon texture.

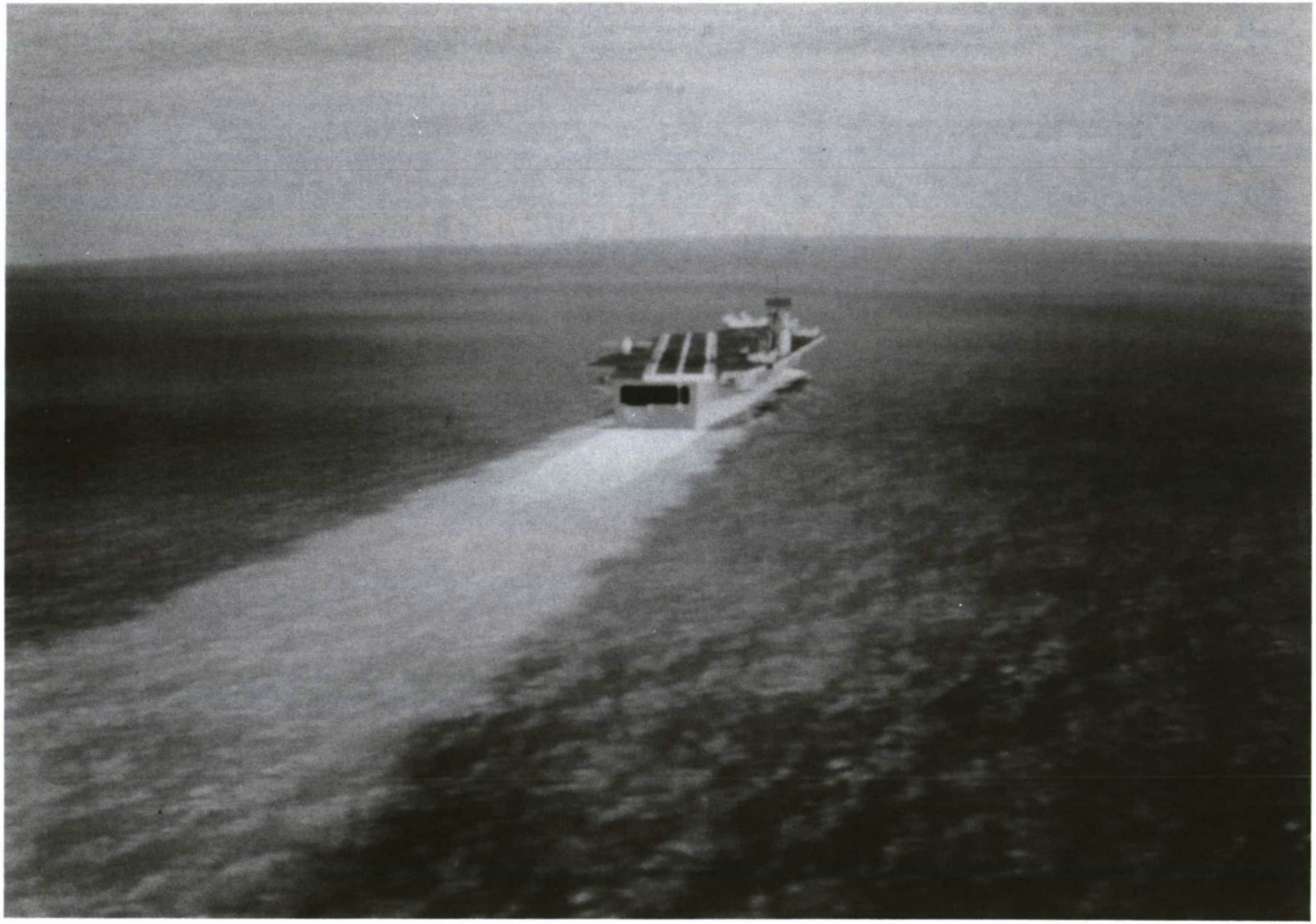


Figure 7

Figure 7 was also photographed directly from the display of a CT6 real-time image generator. The textured ocean as viewed in this photograph is derived from the application of the texture maps shown in Figures 4b and 5b onto a flat polygonal surface. The ocean can be textured in this fashion with a single polygon.

appropriate source photograph and an appropriate image patch subset of a source photo must be executed correctly. Texture mapsets must be composed of patterns with appropriate content and level of scale, and must be applied to the terrain in combination with macro texture and smooth shading to maximize the illusion of realism.

Photo-derived texture promises great potential for enhancing the realism of current and future data bases. Virtually all terrain surfaces in current data bases can now be more realistically represented. In addition, photo-derived texture can also be applied to ocean and cloud models. New capabilities on future CIG systems will make it possible to replace self-repeating patterns with a mosaic of map-correlated discrete texture maps. As the simulation industry becomes more and more fluent with the technology of photo-derived texture, the scene realism and resulting training effectiveness of visual data bases will grow by quantum leaps and bounds.

REFERENCES

1. Burt, P.J., and E.H. Adelson. "A Multiresolution Spline with Application to Image Mosaics." *ACM Transactions of Graphics*, Vol. 2, No. 4 (October, 1983), pp. 217-236.

2. Faintich, M. "A System for All Sources." *Computer Graphics World*, Vol. 10, No. 6 (June, 1987), pp. 91-92.

3. Mandelbrot, B.B. *Fractals: Form, Chance and Dimension*. San Francisco: Freeman Press, 1977.

4. Robinson, J. and S. Zimmerman. "Exploiting Texture in an Integrated Training Environment." *7th IITEC Proceedings*, (November 19-21, 1985), pp. 113-121.

5. Yan, J. "Texture Generation in Real-Time Computer-Generated Imagery." *SIGGRAPH '85 (Course No. 14: High Performance Image Generation Systems)*, San Francisco, (July 22-26, 1985).

6. Yang, S., S. Wood, and D. Wallner. "Generation of Texture Patterns for Realistic Visual Simulation." *8th I/ITSC Proceedings*, Vol. 1, (November 18-20, 1986), pp. 78-85.

7. Zimmerman, S.A. "Applying Frequency Domain Constructs to a Broad Spectrum of Visual Simulation Problems." *1987 Image Conference IV Proceedings*, (June 23-26, 1987).

ACKNOWLEDGMENT

Many individuals at Evans and Sutherland have encouraged the development of the concepts presented in this paper. In particular, the work of Stephen Zimmerman served as a springboard for our effort, and we appreciate his interest and support.

ABOUT THE AUTHORS

L. Charles Clark is the Supervisor of the CT Exploitation Group in the Simulation Division of Evans and Sutherland, and has been involved in the development of CT5A and CT6 data base technology for the past five years. He obtained a B.A. in Geography from the University of Utah in 1980.

Thomas C. Brown is a Software Engineer for Evans and Sutherland's Simulation Division where he is currently assigned to the CT Exploitation Group. He received a B.S. in Computer Science from the University of Utah with an emphasis in medical imagery. His work with medical imagery and computer assisted surgery has been published. Tom has devoted the last ten years to the simulation product line at Evans and Sutherland and has been a major contributor to the development of photo-based texture for the CT6 Data Base Engineering Group.

ACHIEVING REALISM IN SAR SIMULATION

Peter Hunt
Staff Scientist
The Singer Company -- Link Flight Simulation Division
Advanced Products Operation
Sunnyvale, CA

ABSTRACT

Synthetic-aperture radar (SAR) is becoming an integral part of modern airborne warfare and reconnaissance. One of the roles of SAR simulation is to present imagery realistic enough to convey the visual clues needed for training for these tasks. This paper describes a SAR simulation approach that achieves the required realism by modeling the physical properties of the radar illumination process. Central to this approach is the interpretation of all features in the data base as three-dimensional objects. Complex objects can be constructed from several layers of primitives. If ground truth is absent, appropriate synthetic objects are created (houses, trees, roads, and cars) in real time, and are illuminated as normal data base features. To take full advantage of this approach, the format and scope of the data base have been extended to describe complex and moving objects and to include the clues needed to perform real-time synthetic breakup.

INTRODUCTION

Traditional real-beam scanning radars are relatively unsophisticated devices that seldom have resolutions better than 50 feet. The images they produce reveal the grosser details of the earth's surface: the terrain shape, coastlines, and large landmarks such as housing tracts, rivers, or airports. Their poor resolution, coupled with comprehensive data bases, has enabled radar simulators to produce remarkably realistic imagery, resulting in high standards of simulation.

The introduction of synthetic-aperture radar (SAR) has radically changed this situation. The SAR can portray small, tactically significant objects in a recognizable form, and can achieve resolutions better than 10 feet. It is being used for reconnaissance, weapon aiming, and other situations where the identification of small objects is important. However, it is still a radar, and as such creates imagery of a form not immediately recognizable to an inexperienced operator. Simulation, therefore, is still essential in the training of operators to make full use of the new imaging capabilities.

The high resolutions of SAR's have created new challenges for simulator designers, as they try to maintain the standards of realism that the industry has come to expect. In 1983, Szabo(1) presented an outline of the initial steps that Singer-Link had taken to develop a simulator to meet SAR requirements. This paper fills in that outline, describing the design philosophy that has emerged and some of the details of the current design.

The approach described is appropriate for a real-time, multi-role, multi-mode radar simulator (a Digital Radar Landmass Simulator, or DRLMS). It uses as source data the Defense Mapping Agency (DMA) data bases (both cultural — DFAD, and terrain — DTED) and supports all current radar types — SAR, ISAR, real-beam, TFR, tracking, etc. The discussion here emphasizes the storage and processing of the cultural data, rather than the creation of the terrain surface.

TRAINING REQUIREMENTS AND SIMULATION

Radar simulators have two main purposes in training scenarios. First, they are used to teach students, or first-time users, to make full use of the equipment, to recognize malfunctions and take appropriate action, and to learn the conditions under which the best images can be made (i.e., the system limitations). Second, and far more important, they teach the user to recognize the content of the image and relate it to a familiar world and to the current mission. This interpretation of the image, or radar signature analysis, must be fostered even with degraded imagery (all too common for SAR's).

The SAR image gives some unique visual clues that can aid in signature analysis, and these clues need to be reproduced faithfully. They usually revolve around doppler and velocity effects. The main asset of the SAR, however, is the improved resolution, which enables images of parked aircraft, buildings, trucks, etc., to be identified with some degree of success. If a simulator can combine a faithful reproduction of this level of detail with doppler effects and appropriate system degradation, then it can serve as an invaluable training and mission planning tool for the air combat environment.

This paper discusses the approach we use for the generation of high-resolution imagery. We show that where detailed information of the world is available, it should be interpreted both accurately and realistically. If it is absent, then it should be replaced by realistic fiction that neither leads the eye away from, nor directs it towards, the truth. We also show that the model of the world that we use is ideally suited for the generation of effects that are unique to SAR, and its counterpart, the inverse SAR (ISAR).

REQUIREMENTS FOR REALISM

The DRLMS is a real-time device, usually part of a complete aircraft simulator. It takes data from a run-time data base, and converts it into the appropriate radar image. This run-time data base plays a key role in this process, since it must provide all the information content of the image. If visual clues appear that are not derived in some way from this data base, then the image content cannot be controlled, and cannot be relied upon as a training tool. We begin with a brief summary of the source data, and follow this with some thoughts on how a practical run-time data base can be derived from it. This leads to the requirements for the complete SAR simulation system.

Source Data Base

The simulator uses as its primary source of data the DMA digital landmass data base. This data base consists of two parts. The Digital Terrain Elevation Data (DTED) describes the height of the earth's surface, and the Digital Feature Analysis Data (DFAD) describes the cultural features and landforms that sit upon this surface. The DFAD data is stored as a list of feature descriptions, defined either as point features (poles, beacons, isolated buildings), linear features (roads, dams), or area features (forests, complex buildings, housing tracts).

Small areas of this data base are well defined down to the individual structure level, and for SAR purposes can be regarded as ground truth. This level of detail (defined as "Level X" by

DMA) is used for significant navigational aimpoints and target areas. The rest of the world, however, is described at a coarser level of detail (Level 1). Features at this level are more likely to describe a collection of objects. The organization of these objects in the feature is usually left undefined, so any interpretation of this data at SAR resolutions can best be described as fiction.

The simulator must handle both levels of data well. An example of the mix of fiction and truth is shown in Figures 1 and 2. These simulated images show an area that includes both the Level X and Level 1 data. Figure 1 shows a naive interpretation of the source data, with no synthetic breakup. Figure 2, shows default synthetic breakup on the forests and the housing tract in th lower left corner.

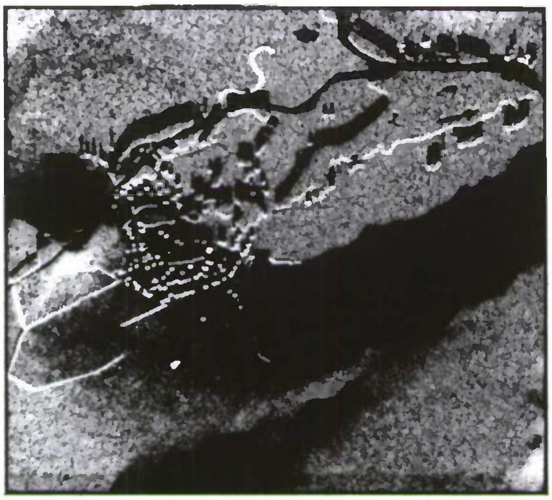


Figure 1 20-FT-RESOLUTION IMAGE ON BOUNDARY BETWEEN LEVEL X AND LEVEL 1 (NO SYNTHETIC BREAKUP)

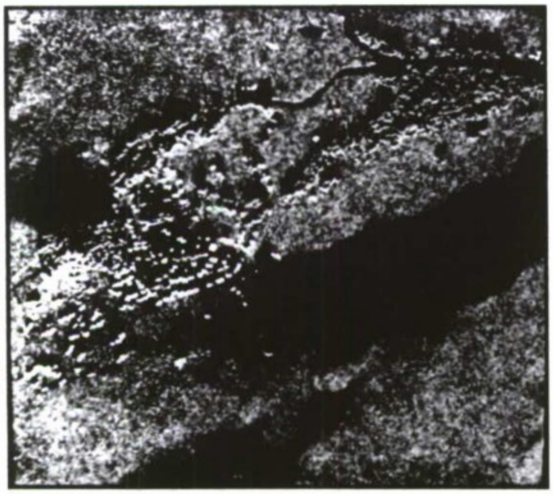


Figure 2 SAME IMAGE AS FIGURE 1 WITH SYNTHETIC BREAKUP

Ground Truth

SAR training needs to concentrate on the detection and identification of both landmarks and targets. A simulator therefore must portray specific, well-defined items in the data base in a truthful way so that the simulated image looks like the real thing. But first, of course, the features have to be included in the data base. The "Level X" data from DMA, where it exists, provides most of the visual clues needed for aimpoint or airport identification. Complex targets, however, such as aircraft or ships, are not found in the initial data base, and cannot always be described in the original format. For complete training, these need to be created (in an enhanced data base format) in such a way as to convey their true radar characteristics (radar signature). The creation of such targets usually requires hand editing.

In order to portray this ground truth effectively, three things are needed. First, there should be enough primitive constructs in the data base to describe any effect that is needed. These primitives should be selected so that any complex structure can be built from them, in a straightforward manner. Ideally, primitives should represent real things, like engine pods or wings, that "make sense" to an editor who may be inexperienced in radar theory.

Second, there should be a close correlation between the data base content and the appearance of a feature in the image (a change in the data base definition of a feature should be reflected in the appearance of the image). This allows for the fine-tuning of a target image.

Finally, ground truth should be interpreted in a consistent manner. Different feature orientations or aircraft altitudes should reflect the basic character of a target, whereas changes in range settings or range should result in correct fading. The target description should also be interpreted correctly for all types of radars (real-beam, SAR, TFR, and ISAR). These requirements affect the simulator design. To achieve this consistency, the full detail and resolution of the data base must be stored and processed for diverse range settings.

Ground Fiction

The ground truth needed for a SAR simulator, with resolutions less than 10 feet, is expensive to produce and store. Most of the world, therefore, is described in terms of multi-structured or homogeneous areas, where only the "mix" of the individual components is specified. The spatial interpretation of these components for SAR resolutions is not defined in the source data base — hence the word "fiction." One of the problems in creating a simulated image is to balance ground truth with this fiction.

This balance is important, because any mission data base is going to contain a mixture of ground truth and fiction. The truth will have been created at a certain level of detail. For valid training, the fiction must have the same level of detail, the same density, and the same characteristics as the truth, since if the truth is seen as somehow different, a student may develop the wrong clues for target or aimpoint identification.

Free-Play

An important requirement in sensor simulation is to allow an operator realistic free-play of equipment during a simulated mission. This means that the operator could fly anywhere in the data base, point the radar anywhere, at any resolution, and expect to get a realistic and convincing image. This capability is a good test of a simulator. If only limited areas of terrain are "allowed" to be examined, then a very rigid training plan must be adopted, flight plans must be rigidly adhered to, and mistakes not permitted. If free-play is permitted, however, it does create a penalty, since it requires that every part of the data base, however coarsely encoded, must produce realistic imagery.

Consequences

These requirements place constraints on both the run-time data base and the simulation system. For repeatable ground truth, the data base must contain all the details of the source data. It must also contain feature descriptions that are easy to edit, and it must then be ready to use, with no complex retransformation after each edit. The free-play facility requires a compact data base, so that the gaming area may be stored on line. Since large areas of the

data base need to be prepared for the gaming area, it is also beneficial if this preparation task is a small one. The simulation system must, of course, interpret the appropriate information in this data base with no loss of fidelity.

We have found that we can meet these requirements with one coherent philosophy. Moreover, the resultant design is not limited to any particular resolution, and can be extended to 1-foot resolution or better.

DESIGN PHILOSOPHY

To satisfy these requirements, we have chosen processing scheme that handles objects rather than properties. The run-time data base has been kept in much the same format as DMA produced it, as a collection of descriptions of objects. This produces a compact data base, which needs very little preprocessing. Yet it still contains the original feature definitions, and so contains the full detail and the resolution of the source data and enhancements.

This object concept is maintained through to the heart of the real-time simulator, where the object is finally "placed" onto the terrain surface and is interpreted as a threedimensional solid body. This is followed by a geometric model of radar illumination, which results in the calculation of a radar echo from the significant faces of the object.

An "object" in this context is not a completely predefined three-dimensional set of surfaces as is found with visual data bases. This would be inappropriate for a SAR image, and would result in an impossibly large data base. Rather, it is a reference to a real entity or landform — a house, road, power pole, or desert — with associated boundary, height, and other radar-significant details. Large homogeneous areas, composed of a mix of different objects, are still defined in terms of their component objects.

If necessary for accurate portrayal, object descriptions can be enhanced with the addition of more explicit details. This process can be continued sufficiently to describe the radar signature of a parked aircraft or a battleship. On the other hand, if the data base does not give enough detail, then objects will be invented in real time, if it is appropriate to do so. Given no cultural data, the scheme will realistically cover the earth with occasional trees and shrubs.

Let's look at how some of the objects are handled in the processing.

Small Objects

When small features (TV antennas, runway lights, etc.) are smaller than a pixel on the radar image, they are retained intact throughout the processing, until they are illuminated. The resolution of the system is such that their size is accurately defined at all times. As a result, when they are illuminated, the returned energy is a realistic interpretation of their true visibility.

Large Single Structures

For high-resolution images, objects may extend over many pixels on the image. Therefore variations in height across the object are noticeable, especially if the shadows the objects create are part of their signature. To interpret these objects correctly, a roof shape or height profile is created just prior to illumination. This shape could be a dome, a sawtooth roof, or the slope of a freeway offramp.

Multi-Structured Features

The simulator is designed only to illuminate objects, or the terrain surface. Therefore any feature described in the data base as a collection of many objects will be broken into the component objects before illumination. This applies to features such as forests, housing tracts, and even parking lots.

OVERVIEW OF PROCESSING SEQUENCE

A brief summary of the main points of each part of the processing sequence follows. These are shown in Figure 3.

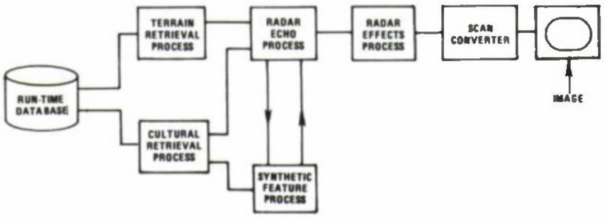


Figure 3 BLOCK DIAGRAM OF RUN-TIME PROCESSES

The *run-time data base*, containing both terrain elevation data and cultural data, uses formats that are based on those of the source data, the DMA DTED and DFAD data bases. The preparation time for this data has been reduced to a minimum, and the data base maintains the original detail and content of the source, with some enhancements. The compact format permits a complete gaming area (say 1 million square miles) to be stored on line at run time, with sufficient detail to satisfy all radar resolutions.

The real-time processing is divided into the usual parts — retrieval, radar echo generation, and radar effects. However, a synthetic feature process has been added that behaves somewhat like an additional retrieval processor.

The *terrain retrieval process* takes the terrain source data derived from DMA (DTED) and retained in the grid format and uses sophisticated interpolation techniques to produce a terrain surface upon which the radar echo processor will place the cultural objects.

The *cultural retrieval process* takes the spatially disorganized cultural data base, still in a list format, and organizes the objects in range and azimuth order. It also subdivides the larger objects so that they may be correctly spread among the pixels on the final image, and simplifies complex shapes that are smaller than the radar resolution. During this process, the original resolution of the data base is retained. Care is taken to prevent any loss of features.

The *radar echo process* places these objects upon the terrain surface (interpolated from DMA DTED data) and gives them the appropriate height. The objects (or parts of objects) are "illuminated" by a radar beam. Instead of a simple radar equation, a more flexible geometric model is used to ascertain the amount of illumination the sides and top of each object receive. This model accounts for shadows and partial shadows created by both terrain and other features, and also the increased energy that can be intercepted from the rays reflected off horizontal reflective surfaces (water, concrete, etc.).

The strength and statistical properties of the radar echo are then computed using the properties of the materials from which the objects are made. Effects of attenuation and antenna gain are also included. Radar echoes are calculated for every object and landform (or part thereof) that has been retrieved by the cultural retrieval process.

In addition to the reflected energy, other parameters are calculated so that the echo from each significant part of the feature may be mapped correctly into the appropriate domain for the

radar effects processing. For SAR simulation, this domain represents doppler/slant-range space. These parameters include sufficient geometric and velocity information to allow layover and the proper characteristics of moving objects to be simulated.

After the energy has been mapped into the doppler/range domain, the *radar effects process* operates on this domain, introducing the effects of the doppler window function, INS errors, range compression, etc. Finally, an image is prepared, using the appropriate radar scan conversion algorithms.

When multi-structured or homogeneous features are retrieved from the data base, the *synthetic breakup process* replaces them with the appropriate objects. It also adds realistic roof shapes to single structures, and molds the terrain surface into mounds.

These, then, are the main components of the simulator. Those that pertain to the processing of culture, the main topic of this paper, will be examined in more detail.

THE RUN-TIME CULTURAL DATA BASE

The data base has already been described as consisting of a list of feature descriptions modeled after the DMA data structure. However, to achieve realism, some degree of enhancement is necessary. In this enhancement, we have tried to stick to the spirit of the DMA as much as possible (especially some early high-resolution specifications, such as the experimental levels V and X).

The Basic DMA Data Base Format

The data base itself contains a sequence of descriptions of all landforms and structures that occur within a given area (called a manuscript for the source data — a much smaller "tile" for the run-time data base). Each landform, object, or homogeneous collection of objects, is given its own description, and is called a feature. The order of these features within the manuscript, or tile, is of no spatial significance, but resolves nesting levels where two or more features overlap.

Each feature description is in two parts. The first part, the header, defines the basic characteristics of the feature — the predominant height, the surface material from which it is made, what it is, etc. The second part defines the outline of the feature. It consists of a string of vertices in geodetic coordinates that define the shape or position of the object in two dimensions. There are three ways that this is done.

Area features have a string of vertices that make a closed polygon defining the outline of the feature. Linear features are assumed to have constant width, and use the vertices to define the centerline of these features with a single open-ended string. Point features are regular circular or rectangular objects, and each has its center point marked by one vertex. For point and linear features, the width, radius, and length are defined in the header.

Data Base Enhancement

To form the run-time data base, the raw data base can be enhanced by adding some inferred or edited data to the header of any of the features. The information is added in the form of microdescriptors — extra packets of data that refine the original header description. Each microdescriptor adds a particular detail or refinement. Many microdescriptors may need to be added to one feature, if it has special significance in a training mission (e.g., a rotating ISAR target). Many of the enhancements can be done automatically. However, those that pertain to complex target generation will probably be hand-edited.

If there are no microdescriptors in a run-time data base, it still will make reasonable images. The real-time processes will substitute the equivalent of default microdescriptors instead. However, during the transformation process a study of the layout of the vertices, or data from other sources, can often supply clues that enable more reasonable microdescriptors to be added automatically. Two common types of microdescriptors are described here.

Filling in the Boundary

DMA features are defined only at the boundaries. For realism, it is necessary to fill in the whole area of large features with detail so that they do not look like bland polygons when displayed in the image. For single-structure features and some landforms, this can be done using a surface to describe the feature height at all points within the feature boundary, thereby portraying a realistic solid structure, with perhaps a domed or sawtooth roof. The surface can be defined either as a simple function (a cone, or dome) or by a lookup table (or pattern) addressed by the appropriate coordinate system (polar or cartesian).

Each feature that uses this function must, however, specify an origin to locate the function, and an orientation and some scaling value to correctly align the surface with the feature boundary. Alternatively, the origin and orientation can be defined by specifying two vertices, one at each end of the feature. This is ideal for defining bridge patterns or strip texture for roads (defined later), since the orientation of such features is important to allow continuity with adjacent features at each end. One microdescriptor can contain all that is necessary for the specification of a roof pattern.

For multi-structured features, a similar function or pattern can be used as the first step in the breakup of the feature into its basic components.

Interpreting the Pattern

The synthetic feature generator needs more than just a pattern that organizes the placement of the component objects. The relative sizes and densities of specific objects must also be defined, and a microdescriptor has been allocated for this purpose. This can sometimes be compiled from source data, since the DMA does provide limited structure densities and tree cover for urban areas. However, unless every forest is to have 50% tree cover, and only a limited number of urban and suburban densities are required, more detail is needed.

One other problem to be solved is that of making a gradual transition from one terrain type to another. Realistic transitions from desert scrub to areas with no vegetation can only be made using several steps. By generalizing the tree and shrub coverage descriptors, several zones can be defined, each with a different shrub size or density.

Resolution Changes

In addition to microdescriptors, the resolution of the data base has been improved, both for vertex definition and for height and width. The DMA data defines point and linear feature sizes in 2-meter steps. As a result, small objects can be given inappropriate dimensions. For example, runway lights cannot be defined smaller than 2 meters high and 4 meters in diameter. Such metal objects are difficult to miss in a high-resolution SAR image. Fortunately, the feature information also defines the objects as runway lights, enabling the problem to be detected and more appropriate sizes to be substituted.

Spatial Organization of the Run-Time Data Base

The source cultural data is usually organized in large manuscripts, often describing all the cultural information in an area one degree square. This unit of area is far too large for a practical run-time data base, so during transformation the cultural

data is divided into smaller areas called "tiles," about 2 arc-minutes across.

REAL-TIME CULTURAL RETRIEVAL

When an image has to be made, those data base tiles that cover the image area must be retrieved and processed, in preparation for the radar illumination task. This processing (illustrated in Figure 4) includes coordinate conversion, fragmenting (or clipping), and sorting, in both range and azimuth.

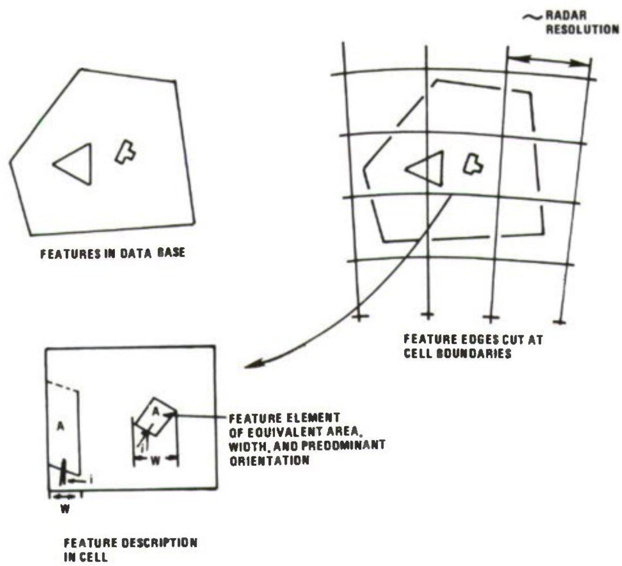


Figure 4 CULTURAL RETRIEVAL PROCESS

In order to organize the data base and make an image, the process creates a polar grid pattern that is laid upon the terrain, over the image area, with the polar origin beneath the aircraft. This pattern divides the earth's surface into "cells" which approximate the resolution of the radar (or a pixel of the image). The grid pattern serves as a convenience to organize the feature data, and the cells act as hooks on which to hand those objects that fall within their boundaries.

The cultural retrieval process retrieves the feature descriptions from the run-time data base and divides these features among the cells they cover. Then each piece of a feature that lands in a cell is simplified so that only the radarsignificant parameters are retained. At this point, the piece of the feature (called a feature element) is seen in terms of the proportion of the cell that it occupies. In this process, no area of a feature is lost, and no feature is ignored.

The cultural data base contains many features larger than a cell. Any cell that is entirely within a feature is defined as having that feature as a background (i.e., it fills the whole cell). The data base nesting rule does not allow more than one feature to occupy the same point, so there can only be one background in a cell (although we have made exceptions to this, as discussed later). There is, however, no reason why there shouldn't be many feature elements within a cell, on top of the background, provided they have boundaries in that cell.

RADAR ECHO PROCESSING

The feature elements have been organized into cells, each of which has a background feature. The cells are now placed upon the terrain surface, and each feature element is given height and "illuminated" with radar energy in a geometrically correct manner.

The radar echo process first calculates the energy that is intercepted by each feature element in the cell. If necessary, this energy is then reduced or eliminated by taking into account the shadows from all feature elements and terrain closer in range to the radar. Each feature element is allowed to have the top surface made of a different material than the sides, so the process distributes this intercepted energy between these two "faces" (see Figure 5). After all the feature elements have been illuminated, any area of the cell background that has not been covered by the feature elements is then processed in a similar way to the feature elements.

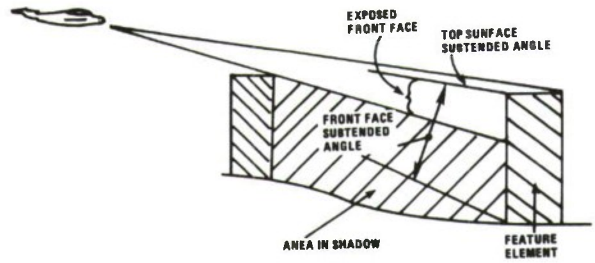


Figure 5 ILLUMINATION OF FEATURE ELEMENT

The calculation of the energy that a feature element or cell can intercept automatically generates the correct range attenuation for the path from the antenna to the object. The remaining task is to calculate the energy reflected back to the antenna. This depends on the reflectance characteristics of the faces (top or sides) of the object and, for many materials, on the effects of the angle of incidence between the ray and the face normal. The resultant echo is reduced by attenuation due to range (return journey) and atmosphere, and by the gain of the antenna.

It only remains for this energy to be mapped into the appropriate domain for the rest of the radar processing.

Transparency

Each feature element has been described as being a solid body that fills the outline it was given. However, in some cases, the object need not intercept all the energy, allowing some to pass on to illuminate objects further out in range. This "transparency" is included to allow objects that resulted from the synthetic feature generation to only partially occupy a cell, or the feature boundary. It enables forests and housing tracts to be realistically simulated even if the synthetic objects are much smaller than a cell (and the resolution of the radar), and allows a little energy to penetrate some distance into these homogenous features.

SINGLE STRUCTURES

Features that consist of a single structure extending over several cells will need some roof structure or surface applied to them to give a realistic radar image. A function or pattern that can define this roof surface has already been described. This information, defined as a microdescriptor in the data base, is carried through the processing to the echo processor. The pattern is referenced on a cell-by-cell basis to supply the height and surface normal.

This process can describe many different features: domed, conical, and sawtooth-shaped roofs, the arch of a bridge, or the hull of a ship (for ISAR). This approach is also effective in generating terrain surface features, from a gentle undulation of the terrain to the irregularities of sand dunes and ice and snow masses.

MULTIPLE STRUCTURES

A very large proportion of features in the DMA data base can be classed as "multistructured," or homogeneous. This class of

features includes forests, housing tracts, parking lots, industrial complexes, etc., and in some cases can be extended to include roads, railways, and airport taxiways. In fact, even the ubiquitous "soil" that seems to cover most of the DMA world is usually a mixture of odd trees, shrubs, and undergrowth.

To maintain the design philosophy, these features need to be interpreted not as a conglomerate single object (except perhaps at long-range map settings), but as a collection of different objects, sitting on some default background. A process is used that can deliver one single object, or part of an object for any point inside the feature. We have already explained that the basic cell in our grid can only have one object as a background within it. However, using transparency, we can partly occupy a cell with an object, and allow a default background to fill the rest.

The previous section explained how a two-dimensional pattern or function could define height at all points within a feature. The task now is to find a way to define not just a height, but a variety of different objects using a pattern. One pattern, by itself, cannot be expected to define very much. First, objects are discrete, not continuous, so a simple interpolation between posts cannot define them. Bringing the posts closer together still creates a limit on the resolution of the radar, something that must be avoided. Second, it would be very difficult to define objects that were not aligned with the pattern grid.

The solution has been to allow a hierarchy of patterns. The first, or master, pattern is used to point to the closest object. Secondary patterns then take over the detailed shaping of the objects themselves.

HIERARCHICAL SYNTHETIC BREAKUP

How can a lookup table or pattern be used to indicate the proximity of an object? There are several approaches to the form this pattern can take. The choice is largely dependent on the architecture used to implement the algorithms.

Using a Surface

To keep commonality with the single structure approach, a surface could be used to define the position and orientation of objects. The pattern could depict sample points of a simple surface, with a ridge line along roadways, and a constant gradient away from the ridge. The height of the surface at any point would then represent the distance from the ridge or road center. This would indicate whether the point was in the road, sidewalk, front yard, house, etc. An additional, cyclic pattern could have ridges perpendicular to this road ridge to represent house centers. The combined patterns would reveal what can exist at any point in the pattern. The surface normal at any point would indicate the orientation of the neighboring house.

This approach is elegant, and allows curved roads and rivers to be defined nicely, but it requires a lot of computation (two interpolations and the orientation of the slopes). It also suffers the drawback that houses may bend around street corners.

A Less Esthetic Approach

A more pragmatic approach is to have each grid post indicate the origin and orientation of the closest significant feature. With sufficient compromises, this can adequately define complex neighborhoods and a variety of different objects. This has the limitation that no two significant features can occupy the same pattern location, and so restricts the distance the grid posts can be apart. However, that restriction can be avoided somewhat by first defining clusters of objects, and then using a second level of the hierarchy to subdivide the cluster, using a finer grid.

The generation of a house using this approach is shown in Figure 6. Examples of a 2-ft-resolution test image are shown in Figure 7. The pattern grid used here has a 20-ft spacing.

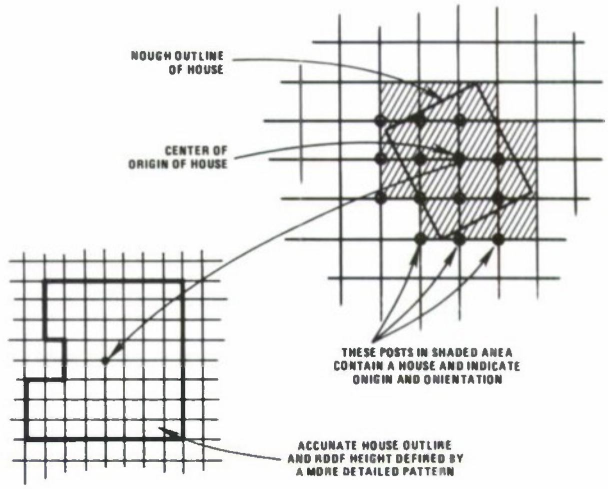


Figure 6 HIERARCHICAL SYNTHETIC BREAKUP

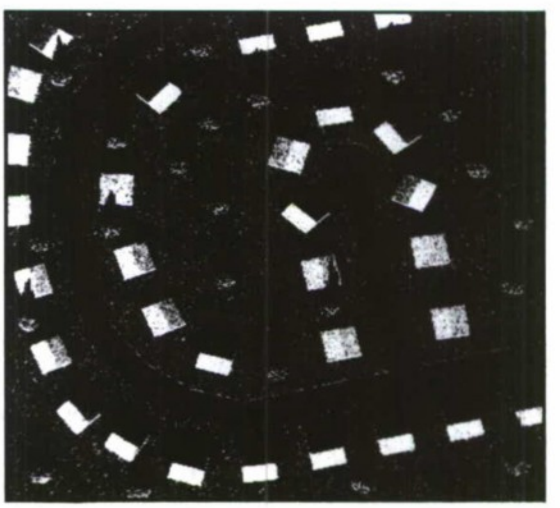


Figure 7 EXAMPLES OF 2-FT-RESOLUTION TEST PATTERN

Beneath the Master Pattern

Once the closest object has been identified and its position ascertained, then the object itself must be defined in more detail. For variety, one of several styles or shapes of the object can be selected. In some cases, the object must be eliminated to satisfy structure density or tree cover requirements. It will be replaced by the default background of this feature.

Once the object type and style have been established, then the height and surface normal of the roof or entry wall are derived, using a simple algorithm or another pattern.

Of course, for a really high-resolution radar or an IR or visual scene, the hierarchical approach can continue. The house pattern could be a house environment pattern that points to the house proper, or to trash cans around the house, chimneys, TV antennas, gable roofs, etc.

Different Range Scales

The master pattern is interrogated, or sampled, on a point-by-point basis. At some range resolutions, the sample points may

skip over significant objects, resulting in an interpretation of the pattern that varies for different range settings, or aircraft positions. The resolution to this problem is to introduce low-resolution data into the pattern. This data enables the appearance of the pattern to show a constant structure density at all resolutions. In addition, it allows the effects of the orientation of the structures in the pattern to be correctly rendered, independent of the resolution. The conglomerate effect of the orientations of the faces of these structures (the "cardinal point" effect) is apparent at resolutions that are far greater than the size of the objects themselves.

The two methods outlined here are designed to handle multiple range scales. The surface method, by its very nature, is ideally suited to a low-resolution radar. The proximity of the nearest house, and its orientation, can be deduced by measuring the slope and height at any point on the surface. In fact, the height itself describes some form a probability of detection.

The alternative, pragmatic method solves the range problem by defining the distance to the nearest significant structure at all parts of the pattern, even far out into unpopulated areas. In addition, all parts of the pattern contain the predominant orientation of the nearest objects.

MORE AIDS FOR REALISM

In the introduction, we claimed that to portray realism a simulator system should be able to describe, within reason, any ground-based feature. Unfortunately, the DMA data base as it stands has two limitations that prevent a full description of certain classes or features.

Height Definition Problems

The first limitation is in the description of predominant height. It has been shown that a surface can be used to vary the height of simple features like ramps or domed buildings. But it cannot handle cases where the height is by necessity independent of the ground. Three feature types show these problems clearly — dams, bridges, and embankments — and these features are often tactically significant.

In these cases, the height can best be described in absolute terms, as height above sea level. An additional thickness term is necessary for bridges, which can have a constant cross-section even if the distance above the ground varies.

One other class of features is given a misleading value from the predominant height. This class includes objects that are supported well above the ground and do not extend down to the soil — for example, elevated water towers and elevated roads (really an extension of bridges). For these cases, the height to the base of the feature is included.

Nesting Problems

The second limitation lies in the nesting rule that is an integral part of the DMA encoding philosophy. This rule states that if two or more features occupy the same point of the earth's surface, then only the last described feature exists at that point.

Many examples can be found where this rule creates difficulties. The most common problem occurs when roads are placed on other structures, such as embankments or bridges. The road is encoded as a zero-height feature, and so will cut grooves in any raised structure upon which it lies. This problem is exacerbated if the road is as wide as or wider than the support structure, since it will completely eliminate the support. Some bridges in the data base are invisible because of this problem.

The data base format has been enhanced to allow three occasions where nesting can be bypassed. First, some objects are allowed to lie on or above other features in the data base. This solves the problem of the bridge and the roads. However, it does result in more processing in the simulator, since both the road and its underlying structure need to be processed for the same point. Just prior to illumination, the overlayed features are placed in their correct positions, and illuminated one at a time.

Second, objects are allowed to exist in addition to the other objects at the same point. The distinction between this case and the previous one can be illustrated by the dam discussed earlier. When the dam has a height less than the elevation of the soil, it does not cut a hole in the soil, but is replaced by it.

The third case relates to the concept of a feature modifier that is not itself visible but modifies the interpretation of the objects it covers. This allows the simulation of such effects as seasonal foliage changes, snow cover, or possibly bomb damage, where the reflectivity or height of all objects must be interpreted in a different way. In addition, some modifier features can raise, move, or rotate objects, such as a rotating ISAR boat.

These modifications to the data base are a natural extension to the idea of storing two-dimensional descriptions of three-dimensional objects. They enable complex objects to be defined by placing many simple structures on one another. For example, a battleship can be defined first as a hull, shaped underneath using an upside-down version of a roof texture pattern. The deck of the boat can then have a multitude of features overlaid upon it. Some may be suspended well above the deck.

Of course, for a SAR or ISAR image, other microdescriptors have to be included to add all the velocity and rotation descriptors that are needed for a realistic simulation. But here again the object-oriented data base and processing can easily pass the motion information along with the object descriptor (header).

PROBLEMS

No scheme is perfect. But the problems that we have found are common to any synthetic breakup approach, whether the breakup is done in real time or off line.

If a synthetic breakup pattern is simply laid on a feature, there will be problems when the feature boundary intersects an object. Trees will be cut in half and sometimes only fragments of houses will be defined. This will be especially evident in the large expanse of the Level 1 data base, where data is unlikely to be edited or corrected.

The approach presented here does have an advantage in this instance. When the data base is ordered, in the cultural retrieval process, a cell can have access to information from neighboring cells. Hence, the proximity of boundaries can be ascertained sufficiently to affect the synthetic object-building functions. A similar process is used for cliff simulation, where the proximity of a cliff must modify the terrain interpolation algorithms in the cliff neighborhood to ensure a terrain discontinuity at the cliff edge.

An alternative solution is to modify the feature boundary during the data base transformation. The outline of the objects in the pattern can be laid on the feature, and the edges of a feature can be moved inwards at the appropriate places to avoid cutting the objects. However, this leads to more complex transformation software, and would require a retransformation of the data base if the patterns were changed.

WHERE THIS APPROACH CAN LEAD US

At first sight, the thrust of this simulation approach seems to have increased the complexity of the data base. In fact, the information included can result in a significant reduction in the

size of the data base, especially in those areas in and around the highly detailed Level X aimpoints since they enable Level X data to be encoded at densities close to those of Level 1.

One appropriate approach for defining urban areas in an accurate but simple way is the idea of a "road strip." This is a synthetic breakup pattern that consists of a simple straight road with a row of houses along each side. The strip can be used to define a complete street environment, using only one feature.

One of the advantages of this approach is that it can make the encoding of an urban area much easier. The data base encoder needs only to define a rectangle around the whole street area (including the houses), and follow this with a specification of the end points of the street, to locate and orient the road strip pattern.

The actual construction of the houses, road, etc., is done in real time. The simulator uses the road strip to build all the objects needed to make the neighborhood realistic (or to match the accurately encoded data elsewhere). Although such a strip could generate great detail, such as parked automobiles, fire hydrants, etc., this may be undesirable, since too much detail in ground fiction is as bad as the lack of detail. Several road strip patterns can be made available to cover the different house densities and junctions or curved roads.

The synthetic breakup philosophy can also be expanded to define convoys, locomotives, battlefronts, guerrilla encampments, river strips (including trees and sand banks), harbors, etc. However, patterns do take up space in the simulator, and only those that are frequently used or save encoding time are worthwhile.

CONCLUSIONS

This paper has described an approach to SAR simulation that gives an accurate and realistic interpretation of targets or aimpoints where they are defined in detail. Since such detailed data base specifications are rare, and are likely to remain so, the approach also puts emphasis on the interpretation of sparesely encoded areas in the data base, creating the same level of detail and realism. We believe that this creates a more effective training environment for the student, since it trains him to reject extraneous detail and helps him recognize the signatures that he must learn to identify under the pressures of combat, from imagery that is often degraded.

In addition, the approach can relieve some of the pressure from data base encoders, who are attempting to surround the truth with other irrelevant truth, often creating unwieldy and highly dense data bases.

Finally, the approach is open-ended, since it can be easily extended to handle any resolution of SAR, ISAR, or even laser radar without major changes.

REFERENCE

1. Szabo, N. "Synthetic Aperture Radar Simulation," I/ITEC Proceedings, 1983.

ABOUT THE AUTHOR

Peter Hunt is a Staff Scientist working in the Research Department at the Link Flight Simulation Division of The Singer Company in Sunnyvale, California. He received his B.Sc. in Mathematics from Birmingham University, and his M.Sc. in Computing Science from Essex University in England. After spending 5 years in the design of real radar systems, he turned to simulation, and has been engaged in the development of new algorithms and architectures for radar simulators for 9 years.

LOW-COST DIGITAL RADAR GENERATOR FOR COMPREHENSIVE RADAR SIMULATION

Dr. George L. Bair
Wayne C. Greaves

Merit Technology Incorporated
5068 W. Plano Parkway
Plano, Texas 75075

ABSTRACT

A new approach to radar simulation is described. It is based upon emerging hardware and software technologies and is suited to many applications including training, engineering analysis, radar prediction, and systems integration. The Digital Radar Generator (DRG) is capable of simulating all air-to-air, air-to-ground, surveillance/command/control, navigation, and air-to-surface (i.e., ocean surveillance) radar modes including high resolution coherent ground map modes and Inverse Synthetic Aperture Radar (ISAR). Low cost is acheived through the use of innovative radar modeling and multiprocessor hardware/software architectures. The hardware architecture evolves from the emerging technologies of: (1) VMEbus, (2) high capacity monoboard computers, and (3) high density RAM boards. The DRG uses Defense Mapping Agency (DMA) standards and special products for data bases to support conventional ground map and high resolution map modes. The paper provides an overview of our design methodologies and concludes with a discussion of a prototype DRG system developed during the previous year and an advanced development DRG system currently under development.

INTRODUCTION

The traditional use for radar simulators has been in training applications. Systems which realistically simulate the capabilities of modern airborne radars for these applications tax the limits of simulation technology. The high complexity of modern radar systems and radar processes drives simulation technology in the areas of processor throughput, landmass data base design, simulation timelines, and resolution. To date there have generally been two approaches taken in the design of radar simulators. One approach uses banks of large disk drives, superminicomputers, and array processors. The other approach uses multiple disk drives and a general purpose minicomputer with special purpose hardware for radar modeling display processing. These approaches, while successful, have the disadvantage of being expensive to develop and maintain and result in a point design which simulates only one specific radar. The resulting lack of flexibility can be a serious limitation given the ease of updating and modifying modern radars. For example, the F-15 APG-63 air-to-air radar incorporates a programmable signal processor which has allowed continual upgrading of the radar's capabilities via software changes. Over the past eight years, only one radar update involved a hardware change; all others were software only.

Maintaining concurrency between a point design radar simulator and the actual radar is becoming harder to accommodate as radar updates become easier to implement. An alternative approach to a point design radar simulator is a reconfigurable radar simulator. A reconfigurable design is one in which the radar performance characteristics of the simulated radar can be specified via an operator interface. This allows a generic radar simulator to be tailored to a specific case. Some of the characteristics of a reconfigurable radar simulator include:

- o Comprehensive Radar Mode Coverage
- o Utilization of Standard Data Bases
- o Simple Set Up and Initialization
- o Standard Input/Output
- o Accurate Registration with Visual Systems
- o Elimination of Simulator Artifacts
- o Accurate Depiction of Candidate Radar Anomalies and Artifacts
- o Maintenance of an Accurate Radar Timeline
- o Rapid Reinitialization
- o Accurate Portrayal of System Failures
- o Adaptability and Flexibility
- o Expansion Capability
- o Full Diagnostic and Self Test
- o Stand Alone or Integral Operation
- o Low Cost, Size, and Weight

Merit's approach to radar simulation is to provide a reconfigurable simulator. Most importantly, our philosophy is to provide a system based on radar theory of operation instead of simply providing a display that resembles a radar. As a result, target detection, tracking, and mapping/imaging reflect the performance of the actual radar system under consideration.

A radar simulator with these capabilities can satisfy an expanded range of applications. In addition to training, these applications include engineering analysis, radar prediction, systems

integration, and radar system design/development. The following section examines these applications in more detail.

APPLICATIONS

The reconfigurable Digital Radar Generator (DRG) provides a new methodology for performing radar engineering analysis. The DRG provides systems engineers a tool to evaluate potential radar designs. Using the DRG, a researcher could postulate a particular set of radar characteristics, input this set to the DRG, and within a matter of minutes be "flying" that radar. In addition, a set of performance parameters can be captured as a candidate radar is being simulated. In this manner, several different designs can be compared and evaluated against a consistent set of criteria.

A new application of radar simulation is radar prediction for mission planning, rehearsal, in-flight replanning, and post-mission debriefing. Previous radar simulator technology did not lend itself to this application. However the reconfigurable DRG, by virtue of its small size and low cost, makes this application very practical. By using the same radar simulator throughout the simulation/ execution/simulation cycle (representing pre-mission, in-flight, and post-mission phases), a pilot would not notice any difference in simulation fidelity. For mission planning, the system may not be required to operate in real-time, and thus could be satisfied by a minimal hardware configuration. Mission rehearsal would use a system similar to that used in a Weapon System Trainer (WST) application to achieve maximum training effectiveness. In-flight replanning functions may make use of avionics system assets, so that the radar prediction function is fully embedded in the on-board computers. The post-flight mission debriefing would use the same simulator as used for mission rehearsal.

The reconfigurable DRG can be a very effective tool for systems integration. In this application the DRG interfaces to other avionics systems, such as a flight processor, through an avionics bus. When interfaced in this manner, the DRG provides a means for testing and evaluating operational flight programs. In the flight processor example, the DRG responds to control inputs such as cursor position or parameter selection and provides data such as track files or height above terrain. Key performance requirements for systems integration include the capability of interfacing to a variety of standard bus interfaces and ease of interfacing the DRG software models with external system input/output.

When used in a design/development application, the reconfigurable DRG can be configured via software to model a conceptual design and then excercised either stand-alone or integrated with a flight simulator. This provides radar systems engineers with a tool to evaluate potential radar designs and provides an entirely new perspective on the cause and effect of radar characteristics on platform and weapon system performance. An additional feature of the reconfigurable DRG is the capability to generate RF/IF/video signals so that the system can operate as a "stimulator" for radar component substitution to provide full operational hardware-in-the-loop simulation. Functionally, the radar simulator must stimulate individual radar components or substitute for particular radar components. In this manner, individual components's contributions to system performance can be isolated, studied, modified, and tested as never before.

COMPREHENSIVE MODE COVERAGE

Comprehensive radar mode coverage is an important capability because current and future military airborne radars continue to exhibit more flexibility and variety of applications than their predecessors. It is imperative that future radar simulators comprehend most, if not all, of these modes. This is true across the entire spectrum of simulators from part task trainers to full weapon system trainers. The mode coverage requirements for the reconfigurable DRG include air-to-air, surveillance/command-control, navigation, air-to-ground, and air-to-surface. These are described in Figure 1.

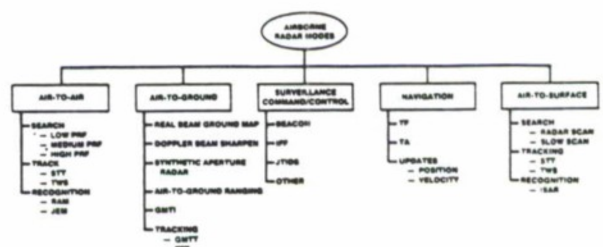


Figure 1. Airborne Radar Modes

Air-to-air modes pose the least challenge to the engineering simulator designer. These modes typically require display of alphanumeric and symbology denoting the air target environment, radar operating parameters, and weapon status. In all modern radars, these systems display synthetic data so that no actual radar imagery is presented to the pilot. If a target is detected, it is represented by a symbol on the radar display.

Surveillance/command/control modes are typically a superset of air-to-air with special mode parameters and symbology. These modes also usually require increased integration with other aircraft communications, navigation, and Identification Friend or Foe (IFF) systems compared to typical air-to-air modes.

On the other hand, air-to-ground modes impose severe challenges to engineering simulators. Past experience has shown that even medium and low resolution real-beam ground mapping (RBGM) modes can present severe tests of processing throughput, database generation, database management, and graphics. With advanced air-to-ground mapping modes now commonly included in new radar systems, the radar simulation requirements have shifted dramatically toward increasing complexity. Doppler beam sharpening (DBS), expanded ground map, and synthetic aperture radar (SAR) modes greatly affect overall the required capabilities of the radar simulation. For instance, consider that a current radar simulator procurement has a requirement to support a SAR mode with a resolution of 2 feet. This has severe impact on database storage, retrieval, and overall processing throughput required to maintain realistic radar processing timelines in the simulation. Navigation modes such as velocity and position updating generally are special operations performed within an air-to-ground mode. For instance, navigators use SAR maps with precisely defined features (such as an intersection of two major roads) to provide position updates to the inertial navigation system.

Air-to-surface modes include ocean surveillance and, in the most modern radars, inverse synthetic aperture radar (ISAR). Ocean surveillance modes are typically scan converted in a manner similar to the air-to-air modes. The display shows symbols at the location of each target. The addition of land/water boundaries is required to accurately model these modes. ISAR is a sophisticated signal processing technique whereby images of the target (usually a ship) are rendered in a range-doppler space. This mode is extremely useful for ship target identification. To simulate it, however, requires the construction of elaborate target models. These target models might require hundreds of individual isotropic scatterers to allow the ISAR simulation to draw a realistic ship image. Realistic images are essential, because ship identification is the major use of ISAR. Once the model has been defined, the simulation is also a function of ship motion and sea state. The complex interaction of these characteristics must be taken fully into account if the ISAR mode is to be modeled properly. Again, proper timelines are important.

DESIGN METHODOLOGY

Merit's philosophy on the design of the reconfigurable digital radar generator is to perform all radar modeling and display processing in software. This provides the system with the capability to reconfigure to model a variety of different radars. This flexibility is the key to satisfying a diverse set of radar simulator applications. The three major innovative design areas are radar modeling software, system architecture, and data bases.

Radar Modeling

Radar modeling and signal characterization are important components for the DRG because only through a full comprehension of the underlying theory is it possible to provide a realistic simulation. Key areas include filtering, detection, parameter estimation, tracking, and classification. Coherent, noncoherent, and other types of filtering are modeled using filter transfer functions. The filter transfer functions provide signal and interference gains.

The model shown in Figure 2 is used by the DRG and has application to all target detection modes. Note that platform, environmental, and target parameters form inputs which are used to compute approach geometry, clutter and interference characteristics. The range/velocity visibility calculations are used to determine whether the target is in a range or velocity blind zone. Radar equations provide an indication of signal and interference power. Filter models are used to compute processing gain against each type of interference and each of the interference powers are modified accordingly. The interference powers are combined to compute a composite threshold. Then, the signal power and threshold are used to determine the probability of detection (this probability of detection is normal for a single dwell). Whether or not a target is detected and displayed depends on the outcome of a random number generated according to the detection probability. Post-detection processing, normally dwell-to-dwell prior to signal declaration, is included.

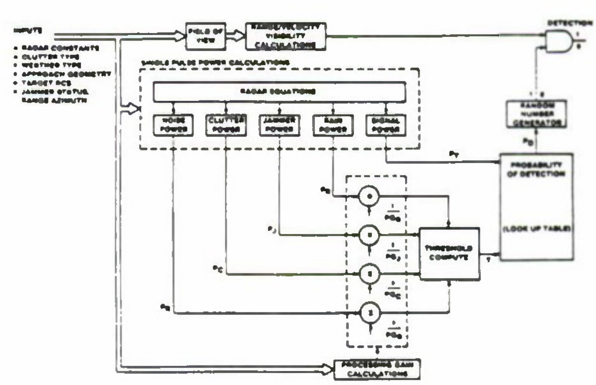


Figure 2. Model For Target Detection

Parameter estimation is used in several of the radar modes to provide radar measurements of range, range rate, azimuth angle, and/or elevation angle to the target for further processing. Our approach to modeling this is to use the precise parameter value plus a random variable with appropriate statistics to

form the parameter estimate. The error statistics include bias, standard deviation, and correlation length (or bandwidth). These statistics are computed off-line and parameterized on the signal-to-interference level.

Our philosophy for incorporating tracking into the various radar modes is to implement the trackers instead of modeling them. This provides a much more realistic indication of tracker operation than models and is actually more straightforward to implement. It provides for a simulation that fully comprehends target and platform encounter geometry and dynamics. These filters can be configured to perform any of the radar modes and apply to both single target track (STT) modes and track while scan (TWS) modes.

Alpha/Beta and Kalman Filters are provided and are illustrated in the functional block diagram of Figure 3. The alpha/beta tracker is a fairly simple nonadaptive filter, which treats each estimate dimension separately. The inputs that define the alpha/beta tracker are the values of alpha and beta for each estimate.

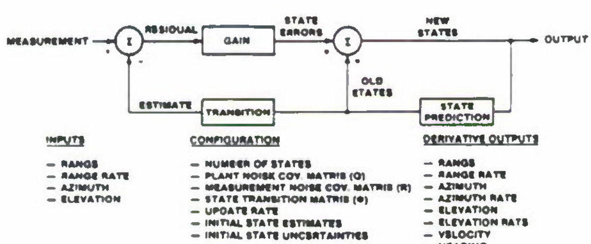


Figure 3. Model for Target Tracking

The Kalman filter is an adaptive filter typically used in modern radars. This can be configured for up to 9 states and is in Cartesian coordinates which is the conventional form. Inputs that define the Kalman filter include: (1) time increment, (2) measurement error covariance matrix which is the statistics of the target parameter estimates, (3) target dynamics model (i.e., plant noise) which describes the anticipated target maneuvers and thus the filter response and steady-state bandwidth, and (4) initial target position uncertainty.

The measurement for the trackers are the target parameter estimates described previously with appropriate error statistics and update rates. These measurements depend on the type of radar mode being implemented. Platform navigation system inputs of position and velocity inputs are also required. The filter runs at the proper update rate and maintains the track file of target states. Derivative outputs are computed from the target states.

System Architecture

The DRG is constructed entirely from components which are compatible with the industry-standard VMEbus. This bus specification has enjoyed a great deal of acceptance and popularity, and hundreds of different boards are now available. Its high performance and suitability for multiprocessor implementations make it an outstanding choice for the DRG. Most of the DRG boards are supplied by Motorola, including state-of- the-art monoboard computers, large RAM memories, and communications boards. Figure 4 provides a representative hardware block diagram of the DRG. A single VME bus as shown will suffice for up to 20 boards; larger systems require multiple interconnected VMEbuses. The processors are Motorola MC68020 with 1 MByte onboard RAM and cache memory. The RAM boards are 16 MBytes. The video frame buffer is dual ported.

The DRG software is written as modules running in pipeline, parallel, and distributed fashion. An example is the implementation of the ground map mode shown in Figure 5 which is primarily a pipeline process.

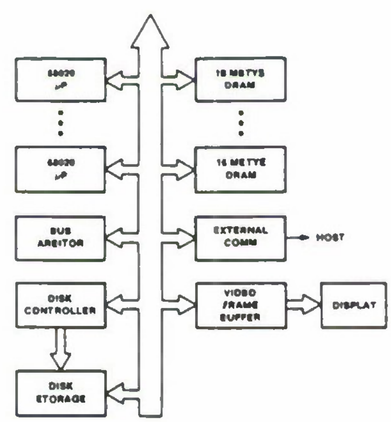


Figure 4. Modular Hardware Architecture

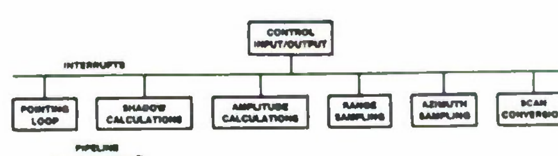


Figure 5. Radar Model For RBGM

The operating system used is a version of the Motorola VersaDOS operating system supplemented by Merit drivers and interprocessor communications software. It has simple, user-friendly interfaces and is a good choice for real-time

applications for the DRG due to low system overhead and the ease with which physical memory on the VMEbus can be allocated and manipulated. It also supports software development and maintenance functions. The use of the VME system as both the development and target machine has the major advantage of not requiring transportation of code from development to target machine. Figure 6 describes the software architecture.

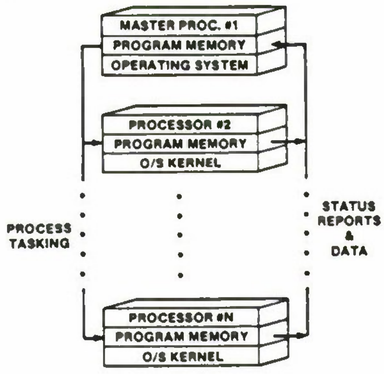


Figure 6. Modular Software Architecture

The programming language for the DRG is "C". This high-order language is especially good for realtime applications because it allows several useful and high speed special functions. It features the ease of use of a high-order language with an execution speed of about 80% that of assembler. Time-critical or repetitive calculations are written in assembler.

<u>Data Bases</u>

The Defense Mapping Agency (DMA) has for years been providing a standard database in a digital format for use in a variety of applications. The Digital Landmass System (DLMS) is composed of Digital Feature Analysis Data (DFAD) and Digital Terrain Elevation Data (DTED). Other more specialized products, such as vertical obstruction files, digital shorelines, and geopolitical boundaries are available from DMA and several other sources.

DTED and DFAD are the standard databases which are generally used as source data for simulations. There are, however, as many ways to process, enhance, texture, modify, and display DTED and DFAD as there are companies working in the field. In addition, DTED and DFAD come in many variations. Essentially, DTED is a regular gridded format. Elevations are posted at 3 seconds of arc (about 90 m) for Level 1 and at 1 second of arc for Level 2. DFAD is not gridded, but defined point, linear, and with area ground features in more of a vector format. DFAD also has Level 1 and Level 2, and these vary primarily by definition of 'point feature' and 'area feature'. Merit uses DTED and DFAD with as little modification as possible. The philosophy is to stack data to create a regular, gridded data base containing both elevation and feature data. For conventional RBGM modes, Level 1 DTED and DFAD may be combined and interpolated to provide sufficient data even for very short range scales. High resolution mapping, however, requires much more dense data than are typically available. Level 1 DTED is sufficiently accurate to establish line-of-sight. The data base for the SAR 'patch' must then be at high enough resolution to support the highest resolution ground mapping modes. Elevation, reflectivity, and orientation are of equal importance. Elevation is required to establish occulting effects, and detailed reflectivity coding is essential to realistic portrayal of radar displays and artifacts. Typically, due to the extreme size of SAR data bases, it is necessary and desirable to define limited areas in training scenarios wherein high-resolution data are maintained. Surrounding areas are maintained in lower resolution and artificially enhanced so that imprecisely designated SAR maps have realistic imagery even in areas where high-resolution data are not available.

IMPLEMENTATIONS

Proof of concept for a (Digital Radar Generator) DRG was accomplished last year with the Merit Technology prototype system shown in Figure 7. The prototype system was designed to systematically address key risk issues of a multiprocessor approach to radar simulation. Current system configuration is six monoboard computers and 10MB of RAM. The key risk areas which have been successfully demonstrated by this prototype include:

- o Processor Throughput
- o Multiprocessor Operating System
- o Run-Time Data Base Management
- o Real-Time Graphics Display

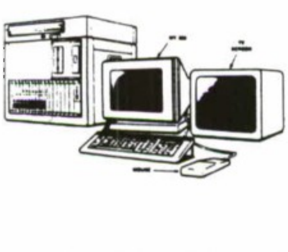


Figure 7. Prototype DRG

The radar modes which have been demonstrated include RBGM, DBS, and SAR. The RBGM mode in Figure 8 has been demonstrated at scan rates in excess of 100 degrees per second. Additional capabilities include varying antenna tilt, sector scan, and a software model of screen phosphor persistence. A simplified Doppler Beam Sharpening (DBS) mode using the Level 1 data base has also been implemented and demonstrated. This high

resolution mode has constant angular resolution (e.g., less than 0.1 degree). Although the level 1 data is rather sparse, texturing algorithms can be used so that the resulting display is representative of operational DBS modes. The prototype system has proven to be extremely reliable, and operates in a standard office environment. Recently, a Synthetic Aperture Radar (SAR) mode has been added and is shown in Figure 9.

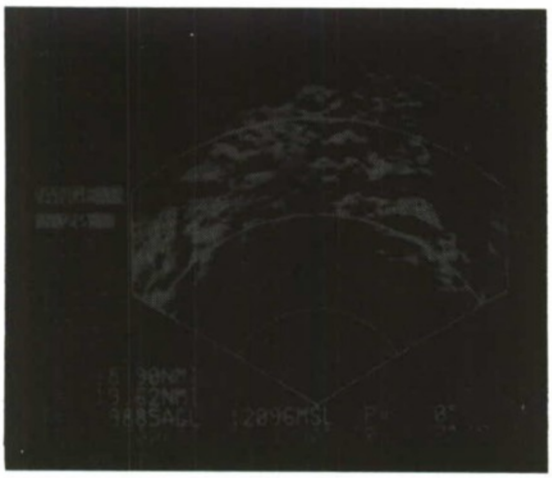


Figure 8. Real Beam Ground Map

Figure 9. Synthetic Aperture Radar Mode

The SAR mode uses high resolution gridded data bases that provide elevation, orientation and surface material information. The range and azimuth ambiguity function of the radar is modeled to achieve remarkably accurate SAR imagery.

Merit is now at work on an advanced development Reconfigurable Radar Landmass Simulator (RRLS) that is based on many of the principles described above. This system is illustrated pictorially in Figure 10 and by the functional block diagram of Figure 11. It features three interconnected VME backplanes, as many as 20 processors, and 100MB of RAM. With mass memory provided by multiple 760MB hard disks, the system can be configured to provide radar simulation over extremely large gaming areas. It will support all major airborne radar modes including air-to-air, air-to-ground, surveillance/ command/ control, navigation, and air-to-surface. This system will truly be a major step in the development of enhanced radar simulators for engineering simulation and will conclusively demonstrate that multiprocessor simulator systems are the cost-effective wave of the future.

Figure 10. Advanced Development DRG

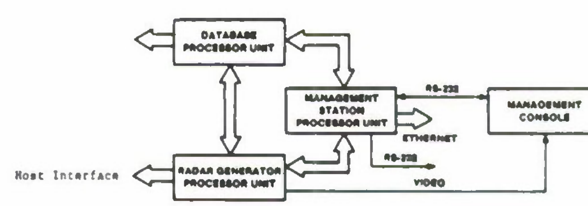


Figure 11. Advanced DRG Block Diagram

SUMMARY

Merit Technology has developed a fundamentally new approach to radar simulation and has designed and constructed a prototype Digital Radar Generator (DRG) which advances the state-of-the-art in radar simulation. By taking advantage of recent developments in multiprocessor hardware and by modeling the radar entirely in software, the Merit DRG provides an unprecedented degree of radar simulation flexibility for training, engineering analysis, radar prediction, and system integration applications.

ABOUT THE AUTHORS

DR. GEORGE L. BAIR received his PhD in Electrical Engineering from the University of Missouri-Rolla. He was previously employed by Texas Instruments where he made significant contributions in signal processing developments for coherent, noncoherent and CW radar systems. His most recent contributions were on antisubmarine warfare radar systems where he conceived, developed, and flight-tested the track-while-scan system for the APS-137 radar, the motion compensation subsystem for the shipboard Profile radar, and the advanced classification aids processor for the Inverse Synthetic Aperture Radar (ISAR). He has authored 23 articles in various professional publications and conference proceedings and has managed a number of system development programs.

WAYNE C. GREAVES has a BS degree from the University of Missouri-Rolla in Electrical Engineering and an MS degree from Southern Methodist University in Engineering Management. He has six years experience on the technical staff of Rockwell International Corporation where he contributed to the analysis, design, test, and implementation of airborne communications hardware and systems including work on VC-137, RC-135, and TACAMO aircraft. He was also responsible for developing real-time application software for a processor controlled PABX system. Prior to joining Merit Technology, he was employed as a Senior Design Engineer for United Technologies Corporation. There he was responsible for developing VMEbus hardware and software products including a 68000 based VME compatible Intelligent I/O board. Mr. Greaves is a Registered Professional Engineer in the state of Texas.

HIGH FIDELITY VOICE SIMULATION SYSTEM

Terry J. Schmidt
McDonnell Aircraft Company
McDonnell Douglas Corporation
Department 251, Building 65, Level 2
P.O. Box 516
Saint Louis, MO 63166

ABSTRACT

The Digital Voice Response System (DVRS) is a totally integrated system which was developed in the Flight Simulation Subdivision of the McDonnell Aircraft Company (MCAIR), a division of McDonnell Douglas Corporation (MDC) at St. Louis, Missouri. The system was designed to simulate Automatic Terminal Information Services (ATIS) broadcasts and Ground-Controlled Approach (GCA) instructions for real-time man-in-the-loop flight simulators and trainers. Consisting of a single printed circuit card integrated into a commercially available personal computer, the DVRS achieves a high degree of realism by digitally recording, during nonreal-time, the voice of an experienced controller or ATIS broadcast (along with associated radio and environmental noise) as a series of messages and then playing back the appropriate message or messages, as selected by the simulation host computer, during real time. In addition to voices, other sounds typically heard in the pilot's environment can also be reproduced by the DVRS. Missile launch, gun fire, engine noise, and aural tones associated with crewstation cautions and warnings are common examples of aircraft sounds.

INTRODUCTION

Simulation technology is rapidly advancing. Traditionally, the effort to increase simulation and training effectiveness has been dominated by research aimed at improving visual and display systems. In order to achieve more realism, however, simulation of the stimuli received by senses other than sight must also be improved. Aural systems technology, for example, requires significant improvement.

Tape recording systems have historically been the most commonly used method of simulating the auditory environment. Due to the nature of the technology, however, time delays (resulting from the process of advancing or rewinding the tape to its proper position for playback) have plagued these systems. In addition, the process of changing the information recorded on the tape involves the difficult task of determining the new locations of the recorded information.

This technology is currently being replaced by computer controlled voice synthesizers or voice record and playback systems. Most commercially available voice synthesizers, however, sound extremely mechanical. Record and playback systems generally produce an unacceptable duplication of the original entry. Errors during the sampling and reconstruction process are a common cause of poor quality. These errors often result from the use of data compression techniques on the sampled data in order to reduce storage requirements.

In an effort to provide accurate simulation of ATIS broadcasts and GCA instructions, a research and development effort was conducted by McDonnell Aircraft Company Flight Simulation. A stringent set of design criteria was established to develop an aural system which avoided the problems that commonly plagued simulations. The effort resulted in a prototype which has since been turned into a production system. The DVRS is currently being used in the AV-8B OFT and WTT, as well as in several in-house simulation programs.

DVRS FEATURES

The DVRS was designed in accordance with the set of criteria developed at its inception. The list of criteria included, but was not limited to, the design and production of a system which:

1) produced extremely authentic voice and tone simulations;
2) experienced no appreciable time delays;
3) could be easily reconfigured and readily applied to a different application;
4) functioned with simulation host computer control,
5) costed less than other approaches.

Accurate tone and voice reproduction was achieved by digitizing an audio input at high frequencies and not compressing the sampled data. The quality of the audio produced by this method reached the quality of conventional tape recorder systems.

The digitized data is stored on a hard disk chosen for its high speed retrieval capabilities. Time delays were further reduced by using a special purpose buffer design. This design allows playback to begin before the entire phrase has been retrieved. Retrieval is completed before the data is needed. The buffer architecture also permits short phrase segments to be combined into longer phrases (concatenated) without time delays between them.

The concatenation capabilities of the system also reduce data storage requirements. For example, three separate phrases can be stored in memory: "glide slope", "above", and "below". These phrases can be combined to form the common GCA instructions: "above glide slope" and "below glide slope". Thus, "glide slope" is recorded only once; but the data is utilized in more than one simulated command. Again, the hard disk capabilities and the buffer design prevent any noticeable time delays even during concatenation.

The software program is designed to allow easy reconfiguration of the DVRS as simulation requirements change. Phrases can be edited by removing words or unwanted audio from the beginning or end of a phrase. Thus, longer phrases can be decomposed and stored as separate shorter phrases.

A phrase can also be re-recorded quickly and easily. For example, "glide slope" can be

recorded as "above glide slope". The new phrase ("above glide slope") will be stored in a manner which is similar to file storage on a computer disk, and the old phrase ("glide slope") will be deleted.

A phrase library and a hole library are maintained by the software during the editing and re-recording processes. The simulation host computer program remains unchanged during these processes, because the new phrases are stored under the same phrase numbers as the original phrases.

The simulation host computer provides real-time control of the DVRS. A standard serial communications link interfaces these two components. The host computer can request either a single phrase or multiple phrases, and the DVRS will produce the audio simulation, without noticeable delay, in the same order as the requests.

The DVRS can be mounted in an IBM PC XT or any compatible. This capability greatly reduces the size and cost of the system.

DIGITAL VOICE RESPONSE SYSTEM COMPONENTS

A block diagram of the system is provided in Figure 1. The system consists of the following:

1) an IBM PC XT or NEC Advanced Personal Computer with a ten megabyte hard disk

2) the CP/M-86 Operating System

3) a Digital Voice Response printed circuit card containing a serial port for communications with the simulation host computer and a discrete port to freeze playback

4) a menu-driven phrase library development and real-time operation software package

5) a headset or microphone and speaker.

The Digital Voice Response circuit card was initially designed to reside in the chassis of the NEC Advanced Personal Computer. Since that time, however, the board has been redesigned to fit into a single card slot in an IBM PC XT or compatible.

PRINCIPLES OF OPERATION

The DVRS must be configured before the real-time output can be used for simulation. The system configuration process initializes the set of blank data files and establishes the set of recorded phrase data files. Once the configuration step has been completed, the simulation host computer and the DVRS can communicate. During real-time simulation, the host computer transmits phrase requests to the DVRS and the DVRS produces an audio output.

System Configuration

The configuration process must be completed before the DVRS is utilized in a simulation application. The hard disk of the personal computer contains a set of phrase files and the phrase library. During system setup, the phrase library is created and maintained. The library contains the phrase numbers, the corresponding track and sector numbers indicating the phrases' storage locations on the hard disk, and the phrase lengths.

In conjunction with the phrase library, a hole library is also maintained. The hole library keeps track of the unused portions of the hard disk. As messages are recorded, the system software searches the hole library to find the next available position on the hard disk for data storage.

During configuration, the record function is used to establish phrase data files. A phrase number is assigned to each recorded phrase. The phrase number is displayed on the DVRS console to provide a correlation between the phrase contents and the phrase number. This information is necessary for selection of the proper phrase in the playback mode. Recorded phrases have a minimum length of sixteen milliseconds, the duration of the audio produced by the 128 bytes of data stored in one sector of the disk. A phrase is recorded by converting the analog audio input to digital data, buffering the data in memory, and writing the buffer to the hard disk. Approximately thirteen minutes of audio can be stored on the ten megabyte hard disk.

During the system configuration process, real-time operation may be simulated by entering the playback mode. Playback during system configuration requires entering phrase numbers on the workstation keyboard rather than via the host computer as in real-time operation. Both options allow a string of up to 127 phrases to be input. The playback process is discussed in detail in the Real-Time Operation section.

After the system is initially configured, the

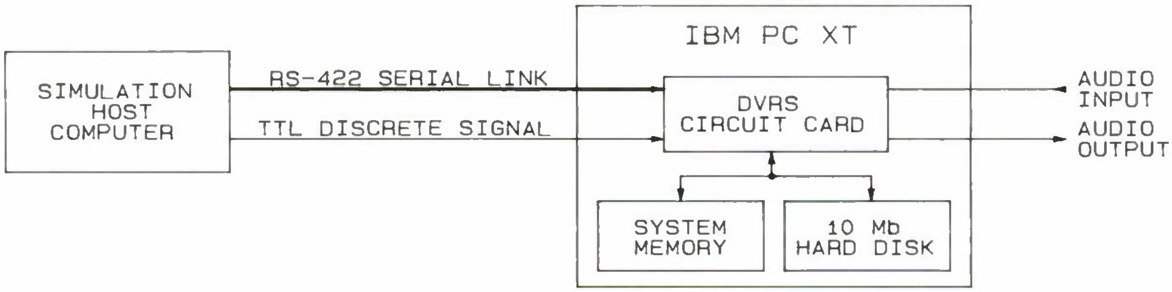


Figure 1. Digital Voice Response System Block Diagram

voice data can be easily altered by re-recording phrases or by using phrase editing. When re-recording an old phrase, the user enters the phrase number corresponding to the data to be changed. The edit function prompts the user to enter the phrase number of the phrase to be edited. The phrase is loaded into personal computer system memory and continuous repetitive playback of the phrase begins. Playback is restricted to the sample data residing between two software pointers, the beginning edge pointer and the ending edge pointer. Phrase editing is accomplished by incrementing and decrementing the beginning and ending edge pointers, respectively, until the desired segment is isolated. Any segment of a recorded phrase, from sixteen milliseconds up to the original length, can be extracted from the original phrase. The edited segment can be stored as a replacement for the original or as a new phrase. As a result, multiple segments of a recorded phrase can be stored, separately for playback in an arbitrary order. If re-recorded or edited phrases differ in length from the original, the new phrase will be positioned on the hard disk according to a set of rules. If the new phrase is shorter than the old phrase it will be positioned at the same location as the old phrase. The phrase library will be changed to indicate the shorter length and a new entry will be added to the hole library. If the new phrase exceeds the space allocated to the old phrase, the new phrase will be positioned at a new location. In this case, the software will create a new hole library entry identical to the old phrase library entry, and the old phrase library entry will be deleted. In the earlier case, the length parameter in the hole library entry will be equal to the difference between the old and new phrase length. Upon completion of the re-recording and editing process, a compact operation can be performed. The compact operation removes holes by placing the voice data in contiguous storage on the hard disk.

Real-time Operation

Once the phrase libraries are established, the system is ready for real-time use. During the real-time, a phrase number (or a packet of numbers) is sent from the simulation host computer to the DVRS via a serial communications link. Upon receipt of the phrase number packet, the DVRS searches the phrase library to determine the hard disk storage location and the phrase length. The proper location on the disk is then accessed and the phrase data is read from the disk. The data is written into random access memory on the Digital Voice Response circuit card until the card's storage capacity is depleted. Any remaining phrase data is stored in system memory on the personal computer. If a group of phrases, instead of a single phrase, is requested by the host computer, the additional phrases are retrieved from hard disk storage in the order of the requests and are stored immediately behind one another, in the personal computer's system memory.

Data is removed from the circuit card's random access memory by the audio control circuitry located on the Digital Voice Response card. As data is removed from the card's memory, the phrase information in the personal computer's memory is transferred into the card's memory until the entire phrase or group of phrases has been loaded. The audio control circuitry makes the digital data available for conversion to an analog audio output.

DIGITAL VOICE RESPONSE SYSTEM DESCRIPTION

The DVRS consists of both hardware and software packages. The development of the Digital Voice Response hardware required designing a multi-functional circuit card which could reside within a personal computer and could produce audio simulation without noticeable time delays. The versatile Digital Voice Response software, which controls the hardware and manages data files, performs many unique functions and contains a menu-driven user interface. Together, they form a totally integrated package.

Digital Voice Response Hardware

The Digital Voice Response circuit card is comprised of several functional blocks as shown in Figure 2. Figure 3 is a photograph of the circuit card. The significant architectural features of the circuit card are the first-in first-out random access memory (FIFO RAM memory), the audio control circuitry, and the serial communications port.

The FIFO RAM memory is used as a buffer for voice samples during record and playback. The buffer is designed around a sixteen kilobyte high speed static RAM. Sixteen kilobytes is sufficient to contain two seconds of sampled data. Buffer control circuitry surrounds the memory. This circuitry generates the RAM address, maintains a counter whose value is equal to the number of bytes in the buffer, and generates full and empty status signals. Additional data enable circuitry allows the buffer's addressability to be time-shared, in 200 nanosecond intervals, between the bus interface and the audio control sections. With the surrounding circuitry, the high speed static RAM is transformed into a high speed, dual-ported, large density FIFO memory. As a result of the architecture, the large buffer consumes only a single address in the personal computer's address map. While recording, the buffer prevents the loss of any data during transfers to system memory. During playback, separately recorded phrases can be played back in a continuous string, because the buffer provides the data to the audio control circuitry in an uninterrupted stream.

The audio control circuitry is responsible for the conversion of voice data between the analog and digital domain. The circuitry also governs data flow to and from the FIFO RAM memory buffer. The audio output is compatible with a headset or an external amplifier. A National Semiconductor TP3051 CODEC integrated circuit buffers the signal, filters the signal, converts the analog input to an 8-bit digital sample, and returns the digital sample to an analog audio output. An eight kilohertz crystal-controlled clock is provided to the CODEC. Incoming audio signals between 200 and 3400 hertz can be accurately reproduced.

A separate optically isolated TTL discrete input allows precise host computer control of the data conversion process during both record and playback. The discrete input enables the system to freeze at some arbitrary instant in time and continue from this exact point at a later time. Playback can also be suspended while the voice

sample is loaded into the buffer, ensuring instantaneous playback upon host computer activation of the TTL input.

In order to retain the availability of the serial communication ports supplied with the personal computer, a serial communications port was designed into the DVRS circuit card. The link is RS-422 compatible, allowing transmission and reception of data over long interface cables. A one megabit per second transfer rate can be achieved. The serial communications port is isolated from the remainder of the board architecture so that it can be utilized for other personal computer applications when the DVRS is not in operation.

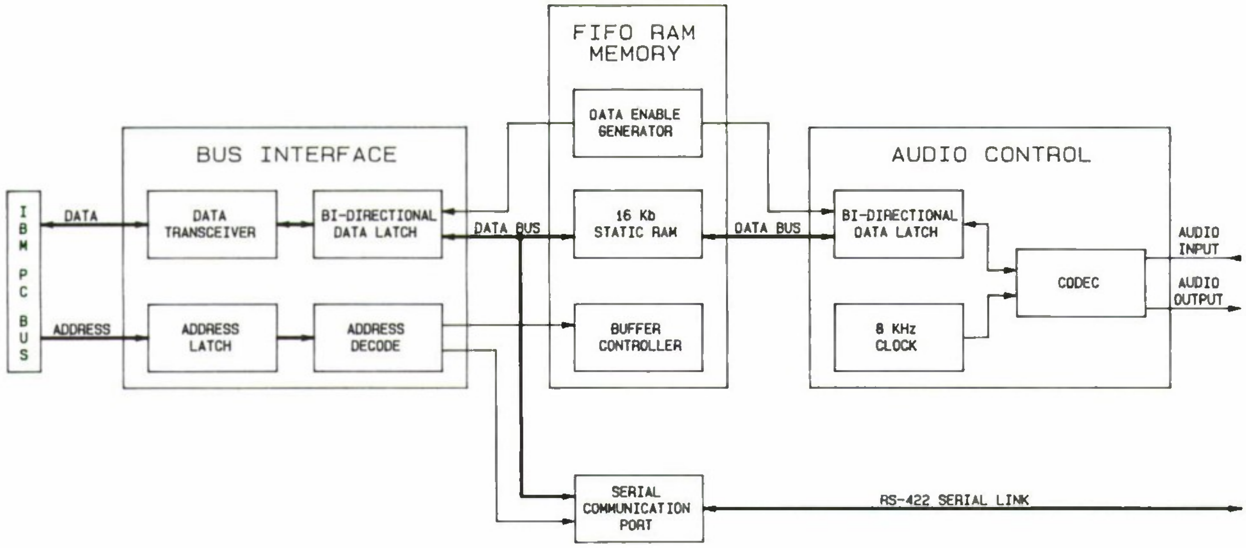


Figure 2. Digital Voice Response Circuit Card Block Diagram

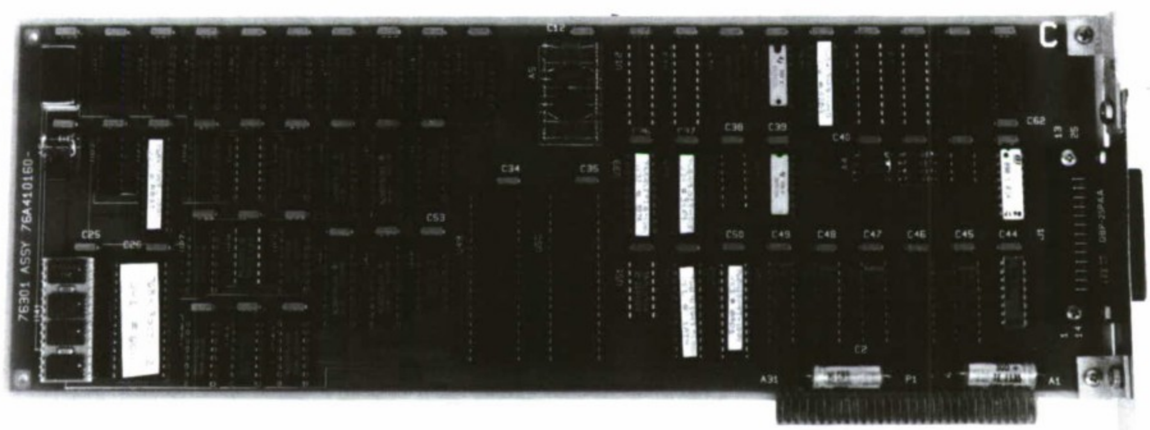


Figure 3. Photogragh of the DVRS Circuit Card

Digital Voice Response Software

The system software was written in Pascal MT+/86 running under the CP/M-86 operating system. Both are products of Digital Research, Inc. CP/M-86 contains operating system calls which perform track and sector I/O to the hard disk. The executable program is completely menu-driven. The most commonly used functions are record, playback, and edit. In addition, other software functions have been programmed into the DVRS. These functions include a complete test of the board hardware, an initialization of phrase and hole library files, and a compaction of hard disk data. Details of many of these functions were presented in Principles of Operation.

CONCLUSION

The Digital Voice Response System is an attractive alternative to tape recording systems, voice synthesizers, and other record and playback devices. Unlike these systems, the DVRS provides a high fidelity simulation with little distortion, and it allows for flexible vocabularies which can be quickly adapted to meet the needs of a new application. In addition, the DVRS is ideally suited for simulation applications because phrases can be played back with little or no time delay. Without a doubt, extremely accurate and highly intelligible reproduction of aural sounds in the frequency range of the human voice can truly be achieved.

ABOUT THE AUTHOR

Mr. Terry Schmidt is a Laboratory Engineer in the Flight Simulation Subdivision of McDonnell Aircraft Company in St. Louis, Missouri. He holds a BSEE degree from the University of Illinois and is a member of the IEEE.

His primary work experience has been in the areas of digital hardware and microprocessor hardware and software. During the last two years he has been working in the Advanced Simulation Technology group of Flight Simulation.

ONE PICTURE IS WORTH A THOUSAND PIXELS - THE GRAPHICAL EDITING OF DIGITAL DATA BASES

J. L. Costenbader
Senior Staff Scientist
The Singer Company, Link Flight Simulation Division
Advanced Products Operation
Sunnyvale, CA

ABSTRACT

The ability to create and modify data bases for digital image generators on graphical devices is not a new technology. Early tablet digitizing programs, however, were cumbersome and difficult to use. Today's advanced graphics workstations have undergone such rapid and significant improvements that it has been difficult for the user community to stay abreast of technology advancements, and the data base generation requirements have advanced almost as rapidly as the improvements in the workstations. Larger data bases, texture, increased data base densities, more complex models, photographic source material, automatic digitizing capabilities, and other features have contributed to the need to marry the new data base requirements to a new generation of workstations. This results in increased productivity, less training time, and better data bases. This paper examines the use of graphical devices in the development of data bases for visual and radar simulation systems. It presents a brief overview of older systems that have served as a springboard to newer technology. Then it examines, in detail, the current state of the art in graphics workstations and modeling systems and how new capabilities are being utilized on these workstations to create data bases for total training systems. Finally, speculation is offered on future modeling systems as workstations continue to improve.

INTRODUCTION

Since the early beginnings of digital image generators and radar simulation systems, some sort of graphical device has been associated with the development of digital data bases for these simulation systems. The effectiveness and productivity of these graphical devices were limited by the inherent capabilities of the graphics hardware and by the software systems developed to exploit them. The desire to use graphics-based systems for modeling was logical, however, since the end product itself was a graphical display. The capabilities and capacities of the early image generators were limited and did not place too great a demand on the modeling systems developed for them. Consequently, the pace of development for graphics-based modeling systems often lagged behind image generator development and sometimes resulted in ill-thought-out implementations. It was not uncommon to see large digitizing tablets leaned up against the wall in a dusty corner because the tablet was difficult to use and not very productive. As the complexities of mission simulation grew and the demand for increasingly complex data bases followed, new and innovative approaches to modeling systems and implementations were soon rendered obsolete by rapidly changing graphics hardware technology.

At Link Flight Simulation, a total systems approach to the use of graphics workstations in the data base development process has been implemented. This approach places the graphics tools at the center of the interactive modeling process and, through careful analysis and evaluation, the best features available in graphics workstations are being used and extended to the modeling process. This paper details this overall systems approach for a graphics-based modeling system, outlines some of the criteria used to evaluate various capabilities of individual workstations, and shows how the best capabilities of the workstations are implemented into a graphical editing process. Some retrospection is necessary, however, in order to understand some of the design decisions that have led to the final approach.

DBGEN SYSTEM

It was clear during the early development of Link's first digital image generator, the DIG I, that a low-cost, easy-to-use, interactive modeling system had to be developed to support the modeling effort for the Shuttle Mission Simulator. This system, DBGEN, was subsequently enhanced and used for the development of the B-52 data bases as well as for early development of models for the Army's SFTS helicopter training program. Nearly ten years later, DBGEN workstations are still in use at Link's Sunnyvale operation and at various user sites throughout the United States. These workstations are still supporting updates and changes to DIG I and DIG II data bases.

The DBGEN modeling system was built around a Tektronix wire mesh X-Y digitizing table and pen. Three-view drawings of the model were taped to the table and digitized using the stylus. Three-dimensional points were communicated to the host computer by touching the same point in two different views of the drawing. Menu commands were generated by touching the stylus to a control area permanently mounted on the tablet. Each control "button" was a half-inch-square box whose location on the table was recognized by the host software. These buttons were arranged in flow chart fashion, the branches of which represented options available to the modeler, and even included a ten-key pad, eliminating the need for a keyboard for numeric entry.

In addition to normal digitizing functions, data manipulation processes were also available. These included mirror, rotate, copy, and place (a library feature). The control flow on the tablet surface also provided for the addition of attributes such as color, intensity, and shading to the various polygons. Verification of work in progress was sent from the host computer to a Tektronix storage screen display terminal. A hardcopy unit was connected to the display terminal so that a permanent record of the model design could be obtained.

Modification of the data base was accomplished by consulting the automatically produced documentation from the original session. This documentation included line drawings and identifying data about all of the features, such as object numbers, face numbers, and vertex numbers. With these identifications, the object or face could be deleted, replaced, or called up on the storage screen. After the changes were made, new documentation was produced so that future changes could readily be made. Once the digitizing was complete, this source data was compiled into a format suitable for use by the real-time system. During this process, separating planes were automatically inserted into the data base.

At the time it was designed (1978-79), this system was the state of the art in digitizing systems. Now, however, more recent technology has revealed the limitations of the early system. First, no model could be created without first making at least a two-view drawing of the model in a scale that would allow easy digitizing. Second, the digitizing-to-viewing process was not interactive, with displays created in the host and downloaded to the storage screen. Menu selection was slow and cumbersome and imbedded on the digitizing tablet. Growth to new options meant significant changes had to be made to the modeling software and the menu overlays on the tablets. Identification of features could not be readily made without the hardcopy documentation. The tablet and stylus were subject to mechanical wear (our original boards have substantial dents in the most frequently used areas of the menus). Once the digitizing was complete, the code had to be compiled, a sometimes lengthy process. Furthermore, the automatic generation of separating planes left no options for the modelers to add their own separation schemes, and the process was sometimes troublesome because of the precision with which the numbers were represented. In spite of this, the system has proved to be reliable, useful, and an effective application of the state of the art.

VISGEN

After a number of years, it became clear that the DBGEN modeling system had outlived its usefulness and a new system called VISGEN was developed. DBGEN was largely used for the development of small areas of a data base and, in the case of the B-52, was a support tool for the B-52 transformation program. The Army's SFTS helicopter program required no transformation software for the development of its data bases, but several highly detailed data bases were designed that were to be entirely hand-modeled. Furthermore, the image generator being used was a significant improvement over previous image generators and required more features which implied costly changes to DBGEN. It was also time to move forward with the state of the art in graphics technology. VISGEN was a design based on the Intergraph CAD terminals and a VAX 11/780 super-minicomputer.

VISGEN improves the throughput, responsiveness, and flexibility of the DBGEN system. Generation of data base objects is by direct interaction with a multiple-view graphics screen, through the activation of high-level functions. The first-level menus are located on the large digitizing tablet and are used to invoke general VISGEN high-level functions. The second-level menu is a CRT-based menu and choice system. The menu network is hierarchically organized so that the user reaches a specific data entry state after interacting with a series of nested menus. The network exposes the user only to the options relevant to the specific task, thus precluding the entry of inappropriate data. Menu choice selections are enhanced by software prompts and feedback messages that occur on the screen.

Several alternatives are provided for defining the geometry of data base features. Such data can be digitized by placing scaled drawings or maps on the digitizing tablet and touching vertex locations with a hand-held mouse. Alternatively, a model can be graphically generated directly on the 25-inch screens by indicating cursor positions in one or more views. For this mode a grid of dots is displayed on the screen to assist in choosing the correct dimensions. Finally, vertex coordinates can be defined by direct input of numeric values via the keyboard. Any of these methods can be used exclusively, or in combination, for any given vertex or combination of vertices.

Many of the standard CAD/CAM features are provided, including zoom, rotate, translate, mirror image, and scale. A variety of graphic display options are also provided. The graphics screens can be divided into four viewing areas to provide simultaneous top, front, and side orthographic views in addition to perspective views with or without hidden lines removed. A solid surface three-dimensional color scene in perspective could also be displayed. This color rendition uses the internal features of the Intergraph equipment and only approximates the functionality of the current Advanced Tactical DIG (ATACDIG) system.

As with DBGEN, the VISGEN output represents source data that must be compiled into a format readable by the ATACDIG real-time system. During this compilation phase, separating planes are added to the model structure. The compiled models are then combined into the gaming area file using a merge process much like a standard linker.

The VISGEN modeling system represents a significant improvement over the DBGEN system. Interactive graphics displays have replaced storage screens, data input is no longer restricted to the digitizing tablet, and changes to the functionality of the system are more easily accomplished. Nevertheless, the system is still hosted by a single super-minicomputer, and when several terminals are operating at the same time there is a direct impact on the response time. The compiler/merge step is still necessary and there is no option for the user to provide his own separating structures. The menu structure is straightforward but inflexible and does not lend itself well to an "expert" mode where the experienced user can bypass some of the prompts.

APPROACHING A NEW SYSTEM

In considering the needs of a new modeling system, some of the lessons learned from DBGEN and VISGEN had to be applied. First, both of the early modeling systems were very much stand-alone products. That is, they were not part of an integrated strategy for the creation of digital data bases for image generation. Although DBGEN was used in conjunction with a transformation program, the graphical editing and transformation programs were completely independent. There was no way to graphically prepare the Digital Landmass System (DLMS) data prior to the transformation operation. Both DBGEN and VISGEN were limited by the capabilities of their graphics systems. Since they were also tied to a super-minicomputer, they became capacity-limited as more terminals were added. The design of the user interface, although good for its time, did not consider abbreviated menus for the experienced user, thus reducing productivity. Neither system had extensive on-line help facilities for the novice user. Both systems required that the workstations be used to make even the simplest changes to the data base. There was no provision for the simple editing of data base attributes via a text-based editor. Separating planes were generated automatically by a separate compilation step, with no user option to specify unique separation strategies. There was also no capability provided for the workstations that allowed a functional emulation of the image generation hardware itself. All attempts at emulation were made through the internal capabilities of the Intergraph equipment.

These and other factors led Link to the development of a new modeling system. Since Link also produces digital radar simulation systems, hooks needed to be provided to include radar data base generation capabilities in the new modeling system. In considering an approach to a new system which would provide good graphical editing capabilities, the type of workstation to be used was the focus of much concern and study.

Graphics Workstations

The selection of a graphics workstation was closely tied to requirements identified in the design of the modeling system. The workstation was not only to be used for the graphical editing of the data bases but also could be used for the process control and configuration management of the entire modeling system. The computer graphics workstation under consideration had to have

a wide range of graphics display functions, including the display of complex three-dimensional objects and terrain surfaces, the display of radar imagery, the display of raw terrain data, the display and overlay of map data, and a highly responsive man-machine interface. Initial investigations into the capabilities of various graphics workstations and terminals soon led to the realization that there were many vendors with excellent capabilities and that the field was changing rapidly. During the course of workstation selection, an evaluation was frequently rendered obsolete by a vendor's new product announcement detailing even more advanced capabilities.

Several Government and trade publications were useful sources of information for putting together an evaluation strategy for these workstations.(1) The data in these publications, combined with the system requirements, was used to create an objective process by which each computer graphics system could be evaluated. A process published in the May 1982 issue of *Computer Graphics World*(2) was used to arrive at a strategy for selecting a workstation:

1) *Identify the desired attributes* — The various computer graphics functions to be performed drive the identification of device attributes.

2) *Rank ordering* — Each of the attributes defined in Step 1 is placed into one of two categories: required minimum and scorable desired attributes. Each feature in the desired attributes category is assigned a relative importance value.

3) *Assign weights* — Each of the desired attributes is assigned a weight representing its relative value with respect to all other desired attributes. The sum of the weights equals 100.

4) *Assign scaling factors* — A unit of measure is assigned for each attribute. Scaling factors, ranging from 0.0 to 1.0, are then established for each desired attribute and represent the percentage of weight to be awarded based on the extent to which a workstation possesses that attribute.

5) *Identify candidate vendors* — Locate as many sources as possible for vendors of computer graphics workstations.

6) *Perform initial screening* — Compare the attributes of the workstations identified in Step 5 with the minimum capabilities identified in Step 2. Eliminate workstations deficient in any of these required features.

7) *Rate the remaining candidates* — Vendors who are fully compliant with the required attributes identified in Step 2 are rated with regard to desired attributes listed in Step 1. The overall rating is then determined by multiplying the scaling factor for each attribute by the weight assigned to the attribute and summing these values over all the features.

8) *Select a system* — Plot the performance ratings of Step 7 against cost. A computer graphics workstation may then be selected based on the best price/performance ratio.

Minimum Requirements. The following minimum requirements were identified in Step 2 (the method of selection is based on the identified overall system design and engineering approach and was arrived at by consensus of the system designers):

— The device shall be a stand-alone workstation

— The workstation shall support virtual memory management

— The workstation shall support multitasking of at least eight user tasks

— The workstation shall include a 60-Hz, non-interlaced, 19-inch color monitor

— The screen shall have a pixel resolution of at least 768 x 1024 pixels

— The workstation shall have at least 8 bit planes of graphics memory

— The workstation shall be provided with a FORTRAN 77 compiler and a DoD-validated Ada programming language compiler/linker

— The workstation shall be able to draw at least 42,000 fully transformed 3D vectors per second

Other minimum requirements dealt with local disk storage, conformity with MIL-STD-1472C, RS-232C ports, display update rates, timing and throughput, and computer interface.

Desired Attributes. The desired attributes for the graphics workstation were arrived at in the same way as the minimum requirements. The method of implementation of each of the desired features was also scored. If the workstation utilized a firmware approach to implement a capability, then the candidate was awarded a 100% score. If the microcode was downloaded at execution time, then the score was 75%. If the function had to be executed in the workstation's main processor and downloaded to the graphics processor, then the score was only 25%. Of course, if the candidate did not provide the capability, there was no score. Some of the desired features listed were:

— 2D graphics transformation capability
— 3D graphics transformation capability
— Hidden line removal
— Smooth shading
— Depth cueing
— Illumination
— Polygon fill (solid color)
— Character generation
— Vector generation
— Display window management
— Graphics libraries
— Line styles
— User-definable function keys

Scoring. Each attribute was assigned a relative weight on a scale of 1 to 10 with regard to its relative importance. For example, graphics libraries were assigned a weight of 7.0 while hidden line removal was assigned a weight of 2.0. The scoring table for these two features is shown in Table 1.

Table 1

Weight	Feature	Scaling Factor
7.0	Graphics Libraries	PHIGS & raster = 1.00 3D GKS & raster = .60 2D GKS & raster = .10 Core & raster = .25 Raster only = 0.0
2.0	Firmware Support Hidden Line Removal	Firmware = 1.00 Downloaded code = .75 SW in CPU = .25 Not supported = 0.0

Rating the Candidates. Requests for quotation were sent to the workstation vendors. Some of the replies were incomplete, some vendors submitted no-bids, and some chose not to reply at all. From the vendors' responses that were complete, compliance with the minimum requirements was established and then the workstations were scored and compared on the additional 27 attributes. Each of the fully compliant candidates was then evaluated with regard to technical performance, cost, and considerations of past performance, market position, etc. Risks for each compliant vendor were also identified. Both a first choice and a second choice of graphics workstations were proposed and final terms negotiated with the vendor of choice.

THE DATA BASE SYSTEM

Prior to the selection of the graphics workstation, an overall system design was conceived for a new data base modeling system. This system consists of both the hardware and software necessary to generate and maintain digital data bases for visual (out-the-window and visual sensors such as IR) and radar simulation. The primary source data for this modeling system is the Digital Landmass System (DLMS) produced by the Defense Mapping Agency (DMA). The DLMS consists of Digital Terrain Elevation Data (DTED), which describes the shape of the terrain, and Digital Feature Analysis Data (DFAD), which describes the cultural features that map onto the terrain surface. Other inputs consist of maps, charts, drawings, and models from which manual enhancements are made. The data bases are largely generated by an automated transformation process and are enhanced and edited as necessary using the computer graphics tools.

System Overview

The tasks to be accomplished are divided into four top-level operations which apply to both radar and visual simulation. These operations are:

— Generation of the initial data base

— Modifications to the data base

— Updating or expanding geographical coverage of the data base

— Validating or verifying generated data

These tasks are allocated to three major functional areas: system management, transformation, and edit. The primary concern of this paper is the interactive graphical editor, but some discussion of the other functional areas is necessary to understand how the graphics workstation has been integrated into the total modeling system.

System management regulates the control over the entire system. It provides configuration management, scheduling functions, and data access control.

Transformation utilizes the DLMS data as its primary source of input. Common transformation performs error detection and correction and converts the data into an expanded format, called expanded DLMS. The visual reformatter uses this data to create polyhedral data for the visual image generator, while the radar reformatter generates multiple resolutions for the radar simulation system.

Edit allows the textual/graphical creation and modification of virtually every data structure in the overall system. Details of the graphical functions of Edit follow. Figure 1 shows a functional flow diagram of the data base system software and highlights those areas in which Edit plays a major role.

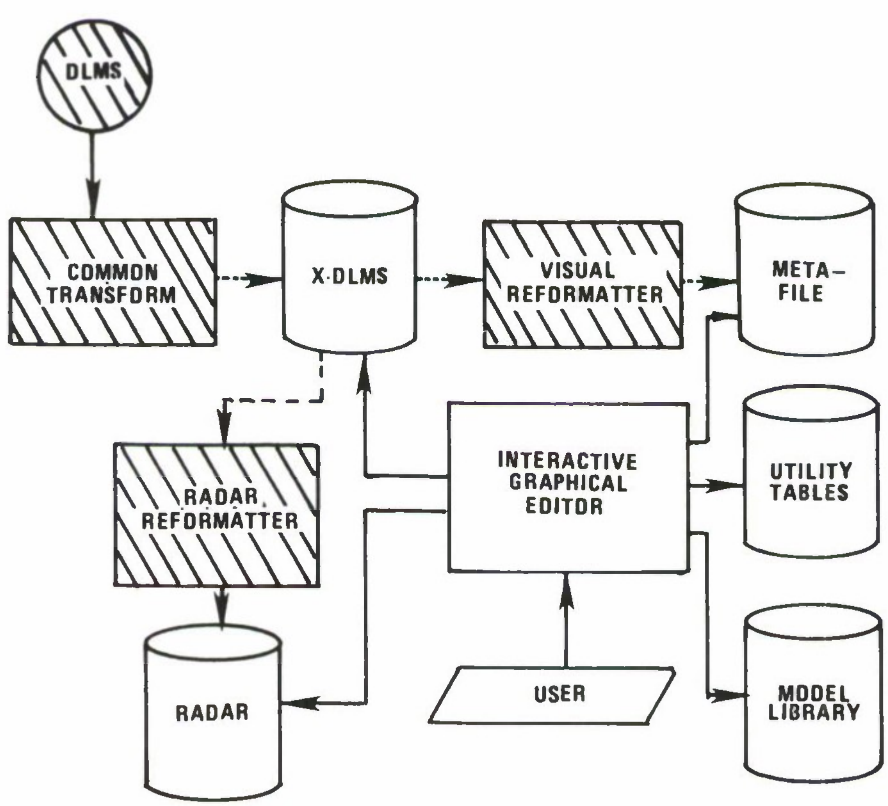


FIGURE 1 INTERACTIVE GRAPHICAL EDITING

One of the primary advantages in using the graphics workstation in so many phases of the modeling system is that it presents the opportunity to design a standard user interface for the modeling system. No matter which major subsystem is being accessed, the basic presentation to the user can be made to be nearly identical, and user functions may be invoked the same way in each subsystem. On the hardware side of the design, a workstation with a powerful internal CPU can be designed to operate in a stand-alone mode, freeing the host CPU resources for CPU-intensive tasks such as data base transformation. The number of workstations in the system can grow easily without significantly impacting the host computer load. Another important part of the design is that many of the functions accessed on the graphics workstation may also be invoked using a text editor on a DEC VT220 (or compatible) terminal. In both the graphical and text modes, commands may be invoked from menus or from the keyboard. The commands available on the text terminal are a subset of those available on the graphics workstaton and exclude those which logically pertain to graphic displays. The user interfaces for the text terminal and the graphics workstation will naturally be different, but the low-level software drivers are, in fact, identical.

Parts of the system are process controllers and use the graphics workstation for relatively mundane displays of processing status, configuration management, production control, and generation of the real-time files for radar and visual systems. The important fact here is that the workstation and associated editors have been integrated into this total system and are not a stand-alone product. These processes are not relevant to the graphical editing of the data bases and will not be discussed here. What is relevant are the graphical editing processes shown in Figure 1, namely, editors for expanded DLMS data, radar data, visual data, and utility table data bases.

Edit Design

As shown in Figure 1, the interactive graphical editor is capable of dealing with a variety of data structures. One of the first design decisions involved the number of editors to create. There were requirements for text-based and graphical editing for expanded DLMS data, radar data, visual data, and utility tables. It was decided that the most efficient approach was to create one editor with slightly different data access routines and modified functionality for each of the different data types. This certainly locked in the notion of the standard user interface. Functions such as window management, gathering of data from the user, and the interface to management would be the same for each sub-editor. Some of the design was driven by the capabilities of the graphics workstation. In order to avoid writing an entirely different set of software for the host computer and the text-based terminals, packages were written to mimic the internal functions of the graphics workstation. The editor also includes an "expert" mode, missing from previous systems, that allows the user to access the various parts of the editor through the use of brief commands rather than relying on extensive user prompts in the "learn" mode. Extensive help functions are also included to aid the novice user. The various editor functions include the following:

- — Modify expanded DLMS (X-DLMS) gridded terrain data
- — Add, delete, and modify cultural features in the X-DLMS
- — Modify and enhance polyhedral data (visual)
- — Modify and enhance radar data
- — Generate and maintain utility data
- — Validate data

Modify Expanded DLMS. When the DMA/DLMS data is processed by the common transformation program, the data is logged in, checked for errors, and blocked into convenient working units. This data is expanded to include additional fields necessary for subsequent processing by the radar and visual reformatters. These fields include such attributes as color, IR codes, etc. This data is an expanded form of the original DMA data and is referred to as X-DLMS data.

The graphical editor is capable of performing many operations on this data, including modifying terrain posts, changing feature attributes, adding new features not found in the original DMA manuscripts, and modifying existing DMA features. This approach helps to ensure that both the radar and visual data bases are generated from the same basic set of data. Target sites and special features necessary for unique training missions need only to be added to the X-DLMS data base by the editor once. This precludes adding features into the visual and radar data bases separately. Issues of separation, clipping, and positioning of features on the terrain model are left to the reformatter programs and do not need user intervention.

The net result is that much of the enhancement to a data base is done prior to the transformation process. This reduces the amount of effort required to insert special features into the data base. The editor also includes the capability to isolate and identify specific features found in the DMA data. These edits and enhancements to the X-DLMS data are reflected in both the radar and the visual data bases after data reformatting. Edits made to radar data or visual data appear in that respective data base only.

Modify Polyhedral Data (Metafile). The editor provides many of the standard CAD/CAM features for editing the visual data base. Some of these features apply further to other subsystems of Edit but generally the requirements for these features are driven by the polyhedral data. Some of these features include:

Display modes:

- Wireframe (visual default)
- Hidden line (radar default)
- Color fill
- Perspective
- Orthographic
- Smooth color shading
- Terrain contour (radar only)
- Overlay terrain with culture (radar only)

Modify modes:

- Modify attributes
  - Texture, color, IR
  - Surface material codes, feature ID's
- Create priority data
  - Automatic
  - Manually generated separating planes
- Create data base hierarchy
  - Entities
  - Groups
- Create/modify geometric data
  - Vertices
  - Faces
  - Objects
- Create construction tools and aids
  - Intersection of two lines
  - Measure distance between two points
- Create/modify utility tables
  - IR
  - Color
  - Switching distance
  - Scene content management tables

Data manipulation

- Copy data elements
- Add new elements
- Delete an element
- Move an element
- Rotate an element
- Scale an element
- Mirror copy
- Slice or clip an element

Perhaps the greatest benefit of the new graphics design is that is has spawned a new approach to the development of the polyhedral model file. In previous systems, it was necessary to generate, during the graphical editing process, a PSP Model File (PMF) (after a hardware processor in the ATACDIG). This PMF had to be compiled into a Compiled Model File (CMF) and then merged into the Gaming Area File as shown in Figure 2. This meant that every time a change was required to the data base, the entire process had to be repeated. This sometimes resulted in significant delays between making the change and viewing it on the image generator.

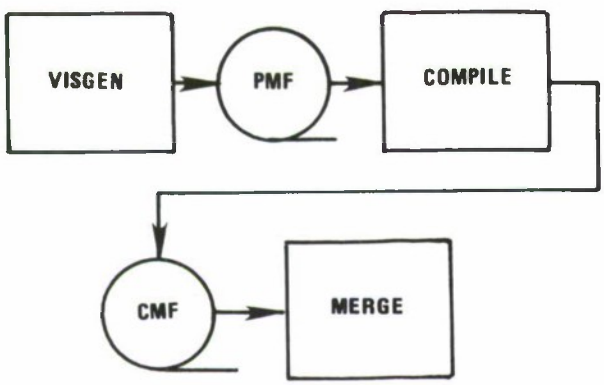


FIGURE 2 PREVIOUS DATA BASE MODIFICATION PROCESS

Borrowing from the display list concept of graphics workstations, this approach was modified somewhat. In most graphics workstations, a display list is built through calls to graphics service routines which allow the display of specific features such as points, lines, and text. The polygonal data base is, in actuality, not very different from a workstation display list and could be accessed by a set of service routines. We have chosen to call this display list a Metafile, and the graphical editor uses a set of calls to Metafile Service Routines (MSR's) to build the polygonal "display list" on the fly, eliminating the need for a merger and compiler (as shown in Figure 3). The Metafile itself is virtually identical to the data structures required by our Modular DIG (MOD DIG), with the exception that the Metafile contains additional descriptive fields to enhance the editing process. This reduces the modify-to-view cycle time. Furthermore, functional emulation of the data base can be accomplished by accessing the Metafile directly and displaying the results using the bit-mapped capabilities of the workstation.

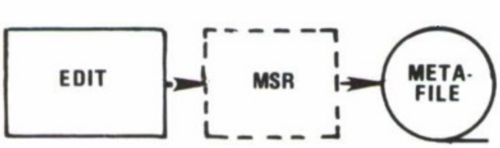


FIGURE 3 METAFILE APPROACH TO DATA BASE MODIFICATION

Modify and Enhance Radar Data. On the radar side, the primary emphasis is to minimize the amount of editing required quired on the several resolutions of radar data produced by the radar reformatter. The on-line radar data base includes high- and low-resolution cultural data, high-, medium-, and low-resolution terrain data, and radar libraries which contain weather systems and targets. With the exception of the libraries, these resolutions are created from the X-DLMS data by the radar reformatter. The ability to edit the X-DLMS data improves efficiency and productivity in that a single change made to the X-DLMS data can be propagated to the various resolutions required by the on-line system. Another benefit, not treated lightly by the design, is that changes made to the X-DLMS data are also reflected in the visual system, enhancing radar/visual correlation.

Radar data may also be edited subsequent to the reformatting of the X-DLMS data. Much of the functionality of this radar editor is similar to that of the visual editor: new culture features may be added, new or replacement feature attributes applied, and target models inserted. Unlike visual data bases, the terrain data and cultural data are kept in separate files and not merged into a single unit. Therefore, the editor provides the capability to overlay the terrain data with the cultural data to ensure that rivers, lakes, etc., conform properly to the terrain contours. The editor also provides the capability to compare the different resolutions of both cultural and terrain data by superimposing one resolution over the other. This provides a visual check which ensures that there are no gross anomalies between the different resolutions. Aside from these overlay capabilities, the editor shares a substantial amount of the visual editor design, which reduces the amount of unique software required and enhances the notion of the common user interface.

Generate Utility Data. Utility data is necessary for both the radar and visual systems, and a method is provided for the creation and modification of this data. Utility data is used mainly by the real-time software on the visual image generator and only a few tables are required by the radar system. Utility data consists mostly of tablets that do not lend themselves well to graphical editing and therefore are generated and modified using a text-based editor on the VT220 terminals. These text-editing functions are duplicated on the graphics workstation using the same software that is resident on the host computer. The utility tables used by the visual system include:

— Switching tables
— Color tables
— Light attenuation tables
— IR tables
— Scene content management tables
— Light/point size tables

Validate Data. It is absolutely necessary to ensure that changes made to any data base are correct prior to releasing the data base for training. Here the graphics workstation pays a big dividend because it is simply not cost-effective to take the image generation systems out of the training cycle in order to validate changes made to the data base. The graphics workstation, however, is not an image generator and functions such as shading and hidden line removal do not necessarily use the same algorithms as the image generator. Consequently, functional emulation of the image generation hardware is used, which maps a snapshot of the data base into the bit planes of the graphics workstation. This emulation also accesses real-time tables for switching distances, color, and IR values, thus aiding in the validation of the utility data as well. The net result is that changes made to the data base can be reviewed on the workstation in emulation mode and it is not necessary to disrupt critical training time on the simulation system for the debug of data bases.

THE FUTURE

It is difficult to assess the capabilities of workstations and their potential use very far into the future owing to the rapidly changing technologies in the workstation environment. During the course of our evaluations, at least two major vendors made product announcements which affected the rating of the workstations. Clearly, some data freeze date must be established during such an evaluation effort.

The demand certainly exists to reduce the costs of developing digital data bases while at the same time improving the fidelity of those data bases. The time required to create data bases needs to be significantly reduced as well. This tends to suggest that automated or semi-automated approaches need to be taken in the digitizing area.

The DMA DLMS data does not necessarily provide up-to-date information, nor does it provide coverage as dense as desired in certain areas. The ideal scenario, in the mission rehearsal environment, is to have reconnaissance photos from a recent mission automatically digitized and inserted into the data base within a matter of hours. Graphics workstations coupled with advanced image scanning systems and driven by expert-based systems may go a long way toward achieving this goal.

Clearly, all three disciplines are maturing rapidly and the future of graphics-based modeling systems will be closely tied to these developments.

SUMMARY

The use of computer graphics devices and workstations for the modeling of digital data bases has evolved roughly in parallel with the development of image generation systems. These graphics devices are no longer stand-alone products but have been incorporated as an integral part of an overall modeling system design. Powerful stand-alone workstations have also allowed decoupling computer resources from the number of workstations used. The advanced graphics capabilities of these workstations have also placed very productive tools in the hands of data base designers and are a far cry from the early tablet digitizing systems. Proper selection of a graphics workstation is critical to the success of the system design. Neither too much nor too little capability in the workstation is cost-effective or efficient. Graphics workstations are here to stay in the modeling environment and, coupled with artificial intelligence and advanced image scanning systems, they will lead to better, faster, and less costly data base development processes.

REFERENCES

1. MIL-STD-1472C, Human Engineering Criteria for Military Systems, Equipment and Facilities.

   Data Sources Hardware-Data Communication, Ziff-Davis Publishing Co., 2nd Qrtr 1986

   "Focus: Graphics Workstations," *Electronic Design*, August 1985, pp. 162-164

2. "Evaluating Raster-Scan Graphics Controllers," Hock Loew & Timothy Nelson, *Computer Graphics World*, May 1982, pp. 65-71

ACKNOWLEDGEMENTS

Special thanks are due to the APO modeling system design team and especially to the Edit design team. It is largely their work that is detailed here. Their talent and foresight have led to a unique design that will set the industry standard for modeling systems.

ABOUT THE AUTHOR

Jan Costenbader is a Senior Staff Scientist with the Singer Company, Link Flight Simulation Division, Advanced Products Operation. As data base architect, his main responsibility is the overall design of data base software tools and digital data bases in support of APO's digital image generation products. He has been directly involved with data bases in the simulation industry since 1982. Holding an Airline Transport Pilot's Certificate with over 5,000 hours of flying time, he has an intimate knowledge of simulation requirements from the users' standpoint as well. Mr. Costenbader received his bachelor's degree in mathematics from UCLA in 1971.

CIG SYSTEM FOR PERISCOPE OBSERVER TRAINING

Peter E. Sherlock, Richard J. Cant
Ferranti Computer Systems Ltd
Bird Hall Lane, Cheadle Heath
Stockport, Cheshire
ENGLAND SK3 0XQ

ABSTRACT

This paper will describe a training system for the cost-effective, real time simulation of periscope visuals using Raster Graphic, Computer Image Generation Techniques. The design is optimised to present high definition target images against a realistic background, with emphasis on sufficient detail and realism to allow periscope observer training in target detection, observation, and classification. A channelized architecture is employed in which target data bases are separately processed to form individual target images. Dynamic background images are generated by a background channel. Unlike conventional approaches, targets and background do not form part of an overall data base; outputs from the channels are mixed together on a priority basis in real time. Target detail is thus maintained independently of overall scene complexity. Smooth edges and motion are sustained by incorporating sub-pixel area antialiasing throughout.

INTRODUCTION

Although similar in many respects to other forms of visual simulation, there a are a number of factors which can strongly influence the design of a system for simulation of the view through a periscope. These factors are a function of the visual environment and the types of skills which need to be taught to satisfy the training objectives of the system. The main training requirements are presented first, followed by a description of how the system architecture and its constituent parts satisfy those needs.

TRAINING REQUIREMENTS

Contacts

A variety of contacts need to be simulated both airborne and seaborne. These should include not only aircraft and ships but also ancillary contacts such as small islands, icebergs and rainsqualls.

In normal operation, an observer will be required to search out and detect contacts, and make a series of value judgements based usually on very short observation periods. The initial impact of the scene on the operator is therefore particularly important and demands a high degree of realism.

Key contact parameters which an observer will be required to assess are its classification, i.e. friend or foe; its angle on the bow (AOB), i.e. its orientation; its range and its speed. All of these parameters require high levels of contact detail for effective training, which should remain consistent at all ranges of observation without distracting and often temporarily confusing transitions occurring.

For example, the determination of AOB often requires an accurate assessment of the relative positions of known vessel structures as a means of gauging a contact's orientation. Not only must the simulator be able to model such fine detail, but it must also be capable of displaying subpixel changes in the image as the AOB changes for a distant contact. Subtle changes in contact shading due to variation in the relative direction of illumination can also affect AOB determination.

For tactical training it must be possible to present the observer with several different contacts simultaneously in the field of view, and a whole variety of contacts in the 360 degree scenario.

Dynamics

The nature of operation of a periscope dictates a requirement for rapid and continuous change in the scene content as the periscope is rotated. The detail in the picture must be carefully controlled to prevent erroneous or missed cueing of the operator.

An operator will often make precise adjustments of periscope position based on closely coupled visual feedback. A very low visual transport delay is thus imperative.

Effective teaching of virtually all observer skills requires smooth motion of contacts, and of the horizon line against which contacts are often measured. The motion must be smooth in both the spatial and the temporal domains. This implicitly demands a system with low aliasing artefacts.

Background

Since a seascape largely lacks specific detail for most of the time, texturing is required in both the sea and the sky to maintain adequate motion cues. The texture should vary in perspective to provide consistent range cueing.

The predominance of sea in the field of view and the fact that the periscope moves relatively slowly with respect to the water establishes a need for dynamic texture to maintain the illusion of realism. This is in contrast to say, flight simulator visuals, where there is sustained perceivable motion of the observer with respect to the background.

A contact is very often obscured unpredictably by the motion of waves in the foreground. The sea should therefore have the capability to obscure the rest the scene. At the extreme, the observation window may be completely submerged.

The appearance and effects of the sea should be definably variable with seastate.

Colour

Variations of colour in a seascape scenario are relatively small compared to other visual simulation requirements. The system should, however, be capable of producing the dominant range of hues in the background for different times of day, and the odd saturated colour such as red and green for presentation of navigation lights.

Special effects

Most periscopes can be operated in two or more magnification powers. These of course must be simulated with the correct field of view.

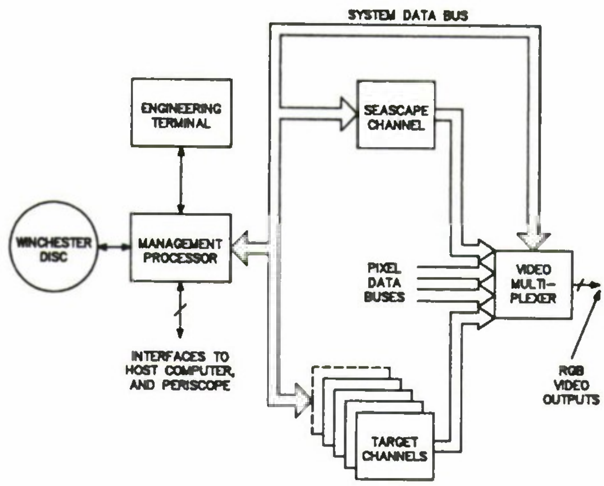
The system should be capable of dynamically simulating the characteristics of bow waves around moving vessels, since these are often used by observers to estimate speed. It should also simulate weather effects such as reduced visibility in fog, and rainsqualls both as contacts and for their effect on visibility.

Target contacts should display appropriate navigation lighting with the correct arcs of illumination.

SYSTEM ARCHITECTURE

The motivation for the chosen system architecture has been the overiding requirement for high intrinsic fidelity as described above. A diagram of the overall system is shown in Figure 1.

FIGURE 1



SYSTEM CONFIGURATION.

In this system, individual contacts and background are initially processed as isolated entities, in separate image generation channels. Target channels generate contact images and the seascape channel generates the background image. This is in contrast to a system in which each channel corresponds to a viewing window.

For this application, such channelization gives an efficient division of work within the overall scene computation task. It provides a nominally consistent execution time without any changes in the displayed level of detail of the contacts. This remains independent of how they are arranged in the field of view with respect to one another, or the background. The specific problem of system overload due to multiple screen coverage is relaxed by building the image of each separate scene element in its own frame-store. This architecture also allows for a much simpler and more efficient control of the system dynamics, by providing independent positioning of each of the scene elements.

The whole system is controlled by a central management processor. Apart from general interfacing and housekeeping tasks, its main roles are to compute and output control parameters for all image generation channels, control the final mixing together of the overall scene, and to control the loading of data bases.

Data bases are initially stored on Winchester Disc from which they are loaded into system memory. The system memory accommodates all target data bases for the current scenario. A further level of data base buffer storage is provided within each target channel, which can hold 2 targets locally. As the periscope is rotated, data bases are transferred at high speed between system memory and target channel under the overall control of the management processor.

Each target channel computes the view on its active model data base, as a function of viewpoint parameters and environmental parameters such as time of day and weather. The perspective image of the contact is built up in a local, frame output buffer and is normally recomputed every frame. The frame buffer contains pixel shade and colour data, and sub-pixel data which allows anti-aliasing to be implemented both when individual images are computed, and when they are finally mixed together in the overall scene.

As will be described later, the seascape channel computes images of the seascape, also stored in local frame buffers. These are produced from a dynamically executed mathematical model, rather than from a stored data base.

Pixel data from all of the channel framestores are passed to the video multiplexer in real time as each raster line of video is output from the system. Each channel is allocated a unique visual priority based on overall proximity to the eyepoint, to give correct occlusion relative to other scene elements. The video multiplexer combines the pixel contributions from each channel, using the visual priority and antialiasing data to compute a final shade and colour for the overall scene pixel. Pixels are converted to analogue form and output to a display system as interlaced fields of raster scan video.

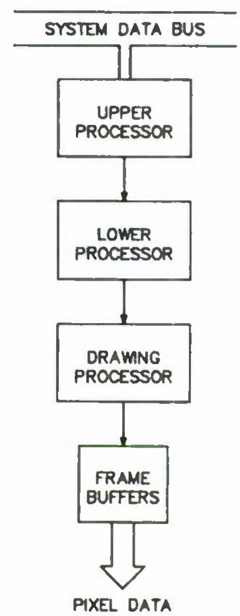
The engineering terminal provides a local user interface to run simple scenario geometry for stand-alone operation, and to execute built-in fault diagnostics.

TARGET CHANNEL

The system architecture implies that there will be a relatively large number of channels if realistic training scenarios are to be possible. It follows that the individual channels must be compact. The current system architecture allows for up to 12 image generation channels including the seascape generator.

The design which has been implemented is essentially a three-stage pipeline as shown in the Figure 2.

FIGURE 2



TARGET CHANNEL CONFIGURATION.

Upper Processor

The function of the upper processor is to perform geometry and shading calculations. It contains a store within which the target data base is held. The required size of this data base was determined by off-line experiments during the early stages of system design. It was found that about 400 visible faces (650 total) is the critical area. To obtain a significant improvement in realism would require many more whilst some targets could not be represented adequately with fewer. As noted earlier there is little to be gained from varying the level of detail in this type of system.

When the periscope is being rotated rapidly it is necessary to change the data base being displayed by a particular channel very quickly. To this end the store is capable of holding two data bases of the specified size, allowing the channel to be reallocated without any dead time.

Processing of the data base begins with the priority sort, done using a binary space partitioning tree structure which is built into the data base during the modelling process. There is thus a minimum amount of work to be done in real time. The ordering of the sort causes polygons nearest the eyepoint to be processed first. This results in more graceful degradation under conditions of channel overload since detail which cannot be drawn is furthest from the eyepoint. It also enables an efficient form of anti-aliasing to be performed as will be noted later. At this point, backward-facing polygons are eliminated from further processing.

Following sorting, the upper processor performs rotation, translation, clipping and lighting computations. The latter include diffuse, direct and specular lighting components, which are calculated on a vertex basis allowing smooth (Gouraud) shading to be implemented. These features are important both for the general realism they impart and to facilitate specific training tasks as discussed previously.

Lower Processor

The lower processor, which is of the same design as the upper, performs line to line interpolation and some of the anti-aliasing calculations. As discussed in the training requirements, a large emphasis is placed on smoothness of movement/ realism which demands a high degree of anti aliasing. The system achieves this by working to an accuracy of 1/128th of a pixel. As an example, the system will display a 1 degree change in AOB for a 600 feet long target, bow-on at a range of 15 kyds.

To meet the processing requirements of both the upper and lower tasks a proprietary microcoded processor has been designed. This was necessary because standard microprocessors are not powerful enough to cope with the required throughput whilst off-the-shelf array processors have inadequate I/O capabilities.

Drawing Processor

The drawing processor writes single pixels and streams of pixels into the framestore and performs pixel by pixel obscuration calculations. These calculations are aided by the nearest first algorithm which allows pixel occupancy contributions to be stored as a simple fraction. The frame stores are double-buffered to facilitate simultaneous writing and display operations.

Typical ship targets cover only 1/10 to 1/4 screen area for normal training ranges so pixel coverage is not critical. Nearer ships are an exceptional case in normal training and can thus be dealt with by allocating 2 channels initially. At even closer ranges the computation rate for the target can be reduced. To maintain smooth temporal changes and a low transport delay, a scrollable frame buffer allows periscope slewing and target movement in azimuth to always continue at the video field rate. This has proved very effective since changes in range and rotation rates (governed by the channel processing time) are relatively slow for ships.

An additional feature of the frame-store subsystem is a high resolution mode in which the line rate is effectively doubled by going into an interlaced mode of operation. The target is drawn double height in the framestore and appears on the screen with its correct height, but twice the vertical resolution. The result is improved performance over a variety of training tasks (especially rangefinding) for distant targets.

SEASCAPE CHANNEL

Objectives

The main objective of the background channel design was to provide a system which would fulfill the training requirements without using an amount of hardware disproportionate to the target channels.

To provide a reasonable level of realism the seascape must be dynamic and detailed. There should not be a discernible "solid" polygonal structure underlying it. Correct perspective implies that, in the more distant parts of the sea, detail should be apparent down to pixel level. It is also important that the motion should appear random and not "repeat" in an obvious way.

The need for foreground waves to obscure targets means that a "flat" texture pattern is insufficient. However, at the same time, the range of each point within the background must be available in order that visibility (fogging) effects can be incorporated. Because of the rapid change of range with screen position near the horizon this must be done on a pixel by pixel basis.

On the basis of these requirements the following System Design idea was developed.

Design Idea

Consider a framestore containing data which represents sea heights at all pixel locations within that part of the screen which is nominally sea (i.e. not allowing for foreground wave effects). Starting from this basis it is possible to compute a surface normal vector corresponding to each location in the store (using neighbouring locations to derive the slopes). From this normal vector a shade can be computed in a similar fashion to that used for a polygonal face in a target. Visibility factors can then be mixed in and the pixel projected to its actual position in the scene using the height value and the range corresponding to that location to derive the angular position.

These calculations are of course much too complex to perform in their entirety in real time, but the judicious use of lookup tables in ram and rom enables a good approximation to be computed in hardware. This implementation retains a surprising amount of flexibility. The effects of seastate, visibility range, sun angle, periscope height and earth curvature are all included.

All this assumes that a suitable set of height data is available, and to compute such a set of data from scratch is obviously a large task. Given that the contents of a frame are likely to be similar to those of the previous one, an alternative approach is to produce a series of frames, each one being an update of the previous. Each frame update would use the previous height at a given location, the heights of neighbours and perhaps the rate of change of height (also stored in a framestore). (It would then be necessary to update the rates of change of height in a similar manner to the heights themselves.) Perspective effects could be generated by varying the coefficients of the updating process with range corresponding to framestore address.

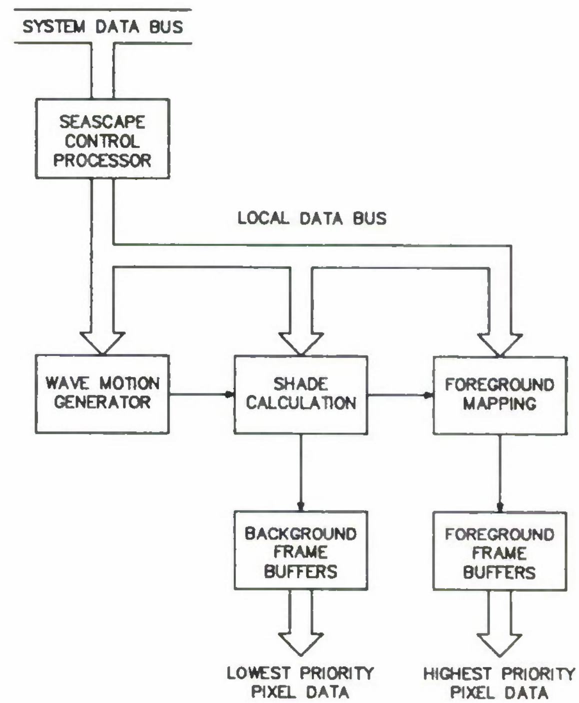
Implementation

The above approach was investigated, the first problem being to find a method of updating which fulfilled the requirements of stability, realism and validity over a variety of perspective ranges.

The solution was found in a pair of filtering operations, one acting on heights the other on their time derivatives. Together these operations recursively solve a linear wave equation, incorporating a low pass filter to prevent an infinite build-up of high frequencies from rounding and truncation errors.

Thus the height fields are made to vary in a wave-like manner. The values of the coefficients used in the filters are held in ram lookup tables and vary with position on the screen to control perspective. The similarity between the two filters allowed a common circuit to be used, a pair of which forms the basis of the wave motion generator. A diagram of the seascape channel is shown in Figure 3.

FIGURE 3



SEASCAPE CHANNEL

To maintain long term stability of the wave motion generator it was found necessary to include pseudo-random disturbances from the controlling processor. This processor executes all the low level control of the seascape channel in response to periscope and environment parameters passed down the system data bus from the management processor.

Height data from the wave motion generator are passed to the shade calculation section which computes shade values for the background seascape in the manner described above. These are written into the background frame buffer. The shades are further processed by the foreground mapping section, which maps pixels to their correct height related position on the screen, and computes their relative obscurations. The final foreground wave shades are written into the foreground frame buffer.

The sky is also written into the background frame buffer. This is a static textured pattern generated in non real time by the seascape control processor.

The frame buffers, and indeed other support hardware are the same as those used in the target channel. Frame buffer scrolling is likewise used to simulate periscope movement.

In the overall system the pixel output data is treated similarly to target channel data. The foreground pixel data is given the highest visual priority and the background the lowest.

CONCLUSION

It has been shown how the analysis of a specific visual simulation requirement can lead to a system solution with a number of fundamental departures from more conventionally adopted techniques. The system described fulfills the objectives of a periscope trainer very cost-effectively for the level of realism achieved, producing an image with complexity in excess of 4000 polygons excluding a dynamic textured background.

The system has been adopted by a number of training establishments internationally and a large library of target model data bases have been produced.

ABOUT THE AUTHORS

Both of the authors are employed by the Training Systems Department of Ferranti Computer Systems Ltd, Cheadle Heath Division in the U.K.

Mr Peter E.Sherlock is a senior systems designer, and project manager for a current CIG system development program. He has been with Ferranti for 11 years and obtained an honours degree in Electronic Eng. from the University of Salford in 1979. During this time he has held a number of design responsibilities in a wide variety of training simulation projects. Mr Sherlock has authored previous publications in Radar simulation.

Dr Richard J.Cant is a senior systems designer primarily concerned with visual simulation systems. He has pioneered the system and software design of the Ferranti CIG Periscope system from its inception up to the present time.

Previous to his association with Ferranti, Dr Cant was a research worker in Theoretical Physics in the University of Manchester and Imperial College London where he obtained his Ph.D in 1979. During this period he wrote nine publications.

AN ADVANCED, LOW COST INSTRUCTOR STATION

Peter M. Tutko
Honeywell Corporation
Flight Simulation Operation
Training & Control Systems Division
13775 McLearen Road
Herndon, Virginia 22071

ABSTRACT

The advanced instructor station design ia based on a systems modularity concept that requires an intelligent IOS whose processing capacity and graphics capability be directly proportional to the training requirements. In identifying the future growth path for instructor station capability, this effort has produced a single IOS design concept that meets this growth potential by isolating the IOS functions and connecting them with a high-speed bus. The identified functions are the graphics engine, an intelligent graphics processor, a cpu for IOS specific functions, an intelligent disk controller, and an intelligent communications interface. This design not only allows flexibility to meet changing trainer requirements, but also gives the designer a flexibility to design to production cost. Application software is written in the Ada programming language and the graphics engine supports a standard software interface library, such as GKS or PHIGS. In short, off-the-shelf hardware (board level) and software components are used to reduce the recurring development effort to the integration of the specific vehicle application.

INTRODUCTION

Perhapa the most common phrase heard in most of today's RFPs for aircrew trainers is "cost-effective training." The cost of the current generation of flight simulators is auch that compromises are often made in simulator performance and training effectiveness due to the high-cost of implementation. While there are several big ticket items that make up the bulk of the cost of a trainer (e.g. visual system, computer system), the overall trainer cost can be attacked by also reducing the cost of other individual subsystems. The Instructor-/Operator Station (IOS) has been identified as one aubsyatem for a coat reduction.

The instructor interface to the trainer is important because, aside from the atudent'a crewatation, all simulator and trainer control is focused around the IOS. Indeed, the student training taak itself originates from the instructor station. Yet the IOS is often not adequately analyzed as a full functional subsystem and usually is run as a background task to the main vehicle simulation within the main simulation computer. This paper preaenta a functional analysis of the instructor interface and of the IOS as a functional subsystem of the trainer in terms of functional responsibility, processing capacity requirements, and graphic drawing capability requirements.

The typical instructor station consists of one or two large color graphic displays accompanied by a set of dedicated switches for display or mode control. The IOS may also include the use of a mouse or trackball, a keyboard, and/or a touchacreen for instructor inputs. Instructor displays are generally arranged in a tree-structured menu of graphic and text pages with the requirement that each display page be accessible by no more than 2 "operator actions." The graphic displays are either all text or a combination of text and graphics (usually two-dimensional). These displays can allow the instructor to insert trainer aircraft equipment malfunctions, view the student's airfield approach technique, monitor cockpit instruments, display cross-country airspace maps and follow the student's groundtrack, and other similar functions.

REQUIREMENTS ANALYSIS

The instructor interface requirements can be roughly divided into three areas for analysis: instructor functions, IOS displays, and processing capability. These areas all affect the inatructor's ability to control the simulation and to evaluate the student's training progreas. In analyzing these functions, current and future training requirements must be taken into account so that any IOS design remains flexible enough to grow with the requirements.

Instructor Functiona

The training instructor is ultimately responsible for designing a mission scenario that will provide a student with the best possible learning environment. To thia end a standard set of IOS functions are typically built into every trainer. These functions include simulator mode selection, map displays, equipment malfunctions, aircraft procedures, cockpit instrument monitors, aircraft environment data, pre-programmed student missions, and simulator maintenance. While it is necessary to examine each of these functions individually and assess their impact on the aimulation task, a aimple summary will be provided here as individual function complexity and implementation may change from trainer to trainer.

Taken as a group, the simulator mode selection, equipment malfunctions, and the aircraft environment data functiona control the operation of the vehicle simulation program. These functions allow the instructor to change the simulation flight mode

(Freeflight, Checkride, Record/Replay, or Demo), change the way the aircraft handles (equipment malfunctions), or change the aircraft flight environment (altitude, weather, fuel, airspeed, etc). The interface of these functions to the simulation program, therefore, will be in the form of control words and data generated by the IOS and input to the simulation. These simulation changes are typically made in between student flights and not "online," i.e. during a training flight.

The remaining functions, map displays, aircraft procedures, cockpit instrument monitors, and pre-programmed missions, allow the instructor to monitor the student's and the aircraft's actions and progress. Generally, the data to drive these functions is output from the simulation program and input to the IOS. It is by monitoring parameters such as altitude, airspeed, rate of descent, engine start data, preflight checklist, and others, that the instructor can use to grade the student's abilities.

IOS Displays

Displays at the instructor station are typically all text or a combination of graphics and text. The text pages may monitor or change aircraft or aircraft environment parameters. The graphic pages are generally maps that show airfield approach or departure routes, aircraft descent profiles, cross country routes with available navigation facilities, or a graphic representation of the cockpit instruments for monitoring purposes. Figure 1 shows a typical text display for selecting an initial conditions set for the simulator.

Present graphic displays are almost exclusively two-dimensional map designs with a static background and have one or more moving symbols dynamically overlaid on top of the background. Displays for landing field approaches are shown as 2-D graphs showing heading and altitude. Figure 2 shows an airfield departure map, including a dynamic aircraft symbol. The trail dots indicate the aircraft track. Tactical displays show the student and accompanying friendly and threat aircraft in two dimensions on a static ground map. The graphics of these displays are generally color line drawings with few, if any, filled polygons.

Future IOS displays will require more three-dimensional representations with a selectable eyepoint for displaying landing field approaches, mission profiles, and tactical encounters. It is not anticipated that these future displays will require 3-D solids modeling or shading from the graphics engine.

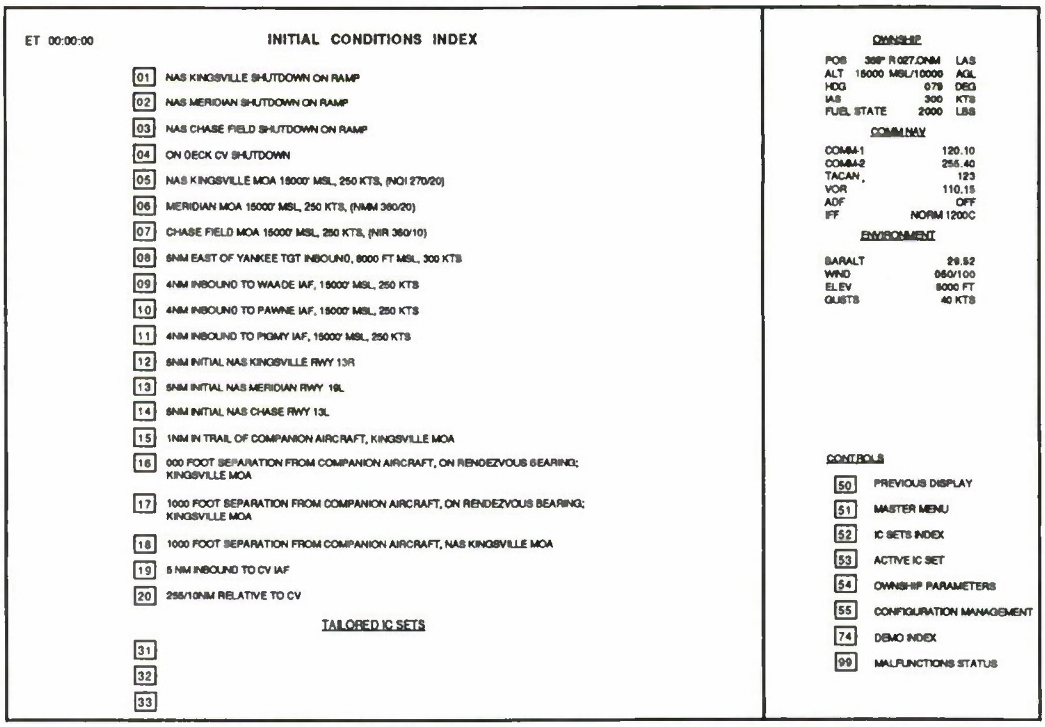


Figure 1.
Typical Text Instructor Display

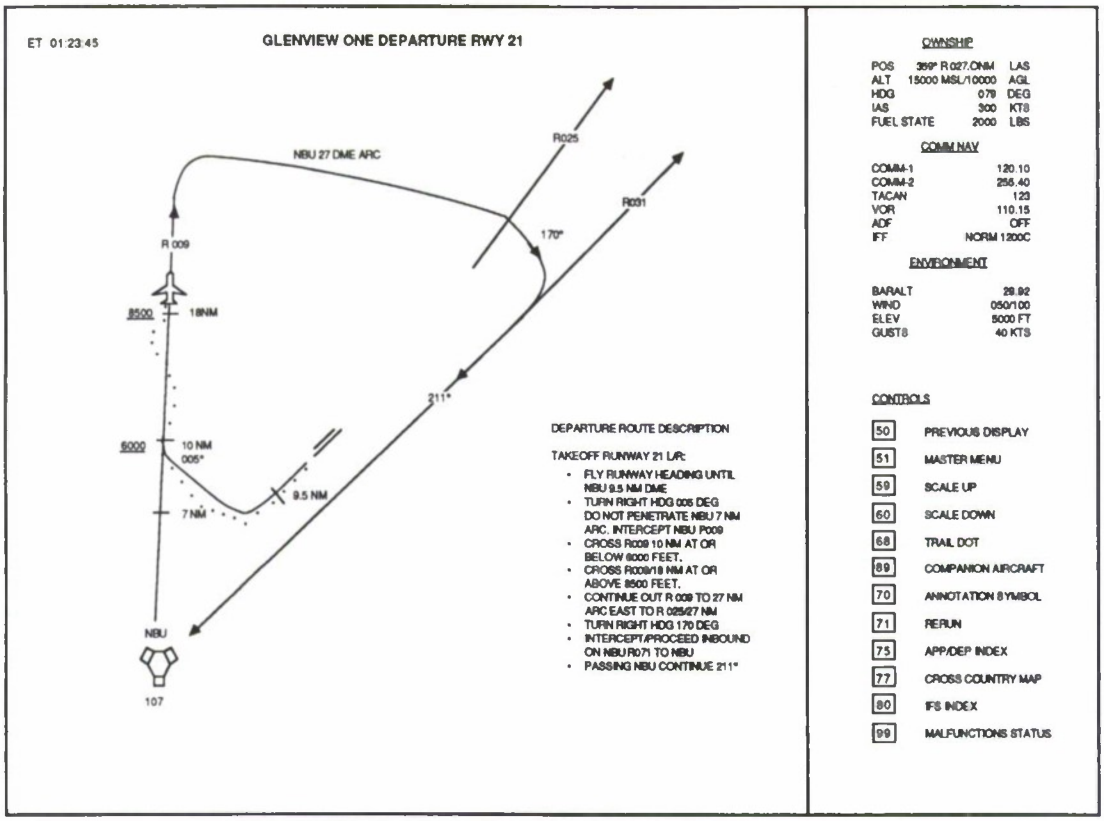


Figure 2.
Typical Graphic/Text Instructor Display

Processing Capability

Most instructor stations presently in the field occupy a part of the main simulation cnmputer. The code and memory dedicated to the IOS may be as high as fifty percent of that used by the entire simulation. Execution of the IOS function is usually as a low priority background task, with a lower priority than aerodynamics or enginea, for example. However, this task can be very I/O intensive and time consuming, depending on the graphics engine and graphics processor used. Most programming for the IOS is relatively straightforward. Any additional aircraft in the simulation (threat or friendly) are run in the main computer as well.

The instructor station of the future will incorporate artificial intelligence algorithms for student monitoring and be required to manage many more aircraft for tactical situations than currently available. Both of these projections will require greater cpu power and more input/output (I/O) throughput.

IOS SYSTEMS DESIGN

The design of any system must be based on an analysia of the functional requirements. In this case, it may be noted that the capabilities growth path for processing and for I/O will be fairly steep. Each functional partition of the IOS must be identified in light of its interface to the main simulation and to each other. With this review, the atructure for the IOS hardware and software becomes apparent.

Functional Partitioning

The IOS as a system is designed by using the major functional processing tasks that will exist in the operator interface. Each task must be examined for its contribution to the overall system processing capacity and to its I/O requirements (both speed and bandwidth).

For the general instructor station, there are five functional subsystems : data communications, dedicated IOS processing, mass storage management, the graphics processor, and the graphics drawing engine. Figure 3 shows a block diagram of the functional instructor station computer.

Data Communications. To process input and output data most effectively, it is desirable to have a dedicated data communications processor. By functionally separating this task (and its proceasing), it relievea the IOS processor from the nuts and bolts of the communicationa link management, thereby increasing its proceaaing capacity for the main instructor functions.

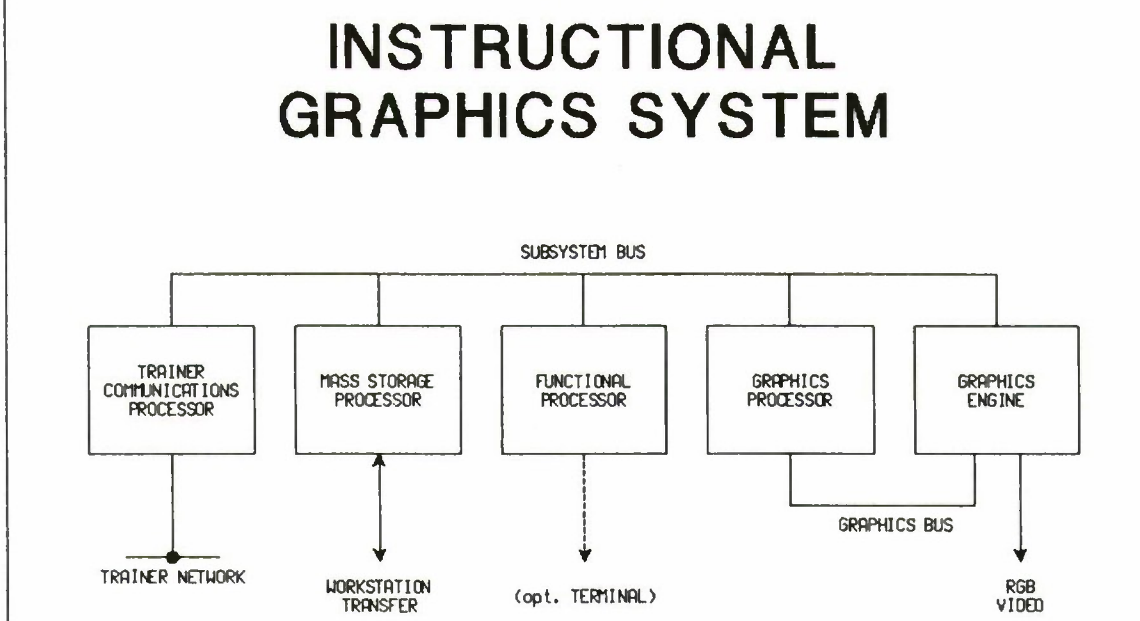


Figure 3.
IOS Computer Block Diagram

Although the communications link to the host may be a high speed parallel bus and indeed this systems design concept will allow any type of interface, the preferred link to the main simulation computer is an IEEE 802.3 based local area network. There are two reasons for this. First, it provides an inexpensive industry standard electrical interface to the rest of the simulation computer system so that computers (or instructor stations) may be changed without affecting the interface. Second, the large amounts of data anticipated for the instructor station will fit within the bandwidth of a 10 MHz serial bus because the information display update is usually low (around 10 Hz).

IOS Processing. The cpu that is responsible for processing instructor interactions and functions should be a general purpose microcomputer, such as an Intel 80386/387 single board computer. Minicomputers (or a part of the main simulation computer) have traditionally been used for this IOS processing. However, the selection of this microprocessor affords similar processing power at a greatly reduced cost.

All IOS software will be written in Ada. Not only is this language required by the DoD, but a modular software system may be designed, and more importantly, reused from contract to contract.

Mass Storage Management. In a typical instructor station, there may be between 150 and 300 separate display pages. Obviously, these pages will need to be stored offline until needed and then be quickly retrieved to facilitate rapid page changes. An ideal disk controller would provide a high level interface to the IOS processor, providing the page data with only a single command. A high-speed disk or a caching disk controller will be adequate.

Graphics Processor. The graphics processor may be another general-purpose micro (identical to the IOS processor) or be a specialized processor within the graphics engine. Regardless of its location or type, the graphics processor should bear the brunt of graphics display list management, segment rotation and translation, and other data necessary for the graphics engine. It is important that a separate processor be used for this task and not shared by the IOS processor. The IOS software program (and programmer) need only be concerned with the IOS specific functions and will often require the full processing capacity of its computer. Processing of these graphics functions is very time consuming.

The graphics vendor should provide the graphics processor and its associated software. Indeed, selection of a graphics vendor will be based in part on its graphics processor and processing algorithms.

Because the graphics capabilities will expand in the future, software support should exist in the form of a standard graphics interface library, such as GKS or PHIGS. With this type of standard library and standard graphics interface, the graphics engine may be swapped out and upgraded at a future date without significant impact on the rest of the IOS system.

Graphics Engine. There are many high-powered graphics engines available in the industry that can provide many CAD/CAE features not necessary in an IOS. Choose one that meets the trainer requirements. Current aircrew trainers require a raster graphics display with 1280x1024 pixel resolution and 256 simultaneous colors.

The interface between the IOS processor and the graphics system should be a standard parallel interface such as a DEC DR11-W. This opens the door for standard, off-the-shelf software drivers and broadens the choice of graphics engines. In the future, should the graphics engine need to be upgraded or changed, chances are very good that the next graphics system will support this interface.

Inter-Function Communication

Because the future processing capabilities in each area are great, it is reasonable to assign a single, specialized processor to each hardware function, as shown in Figure 1. In this way, growth in any or all functional areas may be increased without affecting the overall system structure.

Now that each hardware function is being processed intelligently and in parallel, the communication between the processors must be handled effectively, in hardware if possible, for greater throughput. It is desirable to select an industry standard, parallel bus that addresses this distributed processing issue. This bus is Multibus II. By choosing this standard bus, off-the-shelf interface boards and computers can be used to decrease cost and increase the hardware functionality by distributing the processing. When trainer requirements change in the future that affect any of the processors, a more appropriate processor may be exchanged for the old without significantly affecting the rest of the processors. This capability will save the engineer from the need to redesign the system for every new contract.

Most inter-processor communications should use the Multibus II message passing function of the bus. Commands to the IOS processor for mode control, graphic parameter passing to the graphics processor, and page retrieval information to the disk processor all can use messages, thus freeing most of the bus bandwidth (40 Mbyte/s) for block data transfers, such as new display page information from the disk or new aircraft environment data from the communications processor.

DESIGN TO PRODUCTION COST

As stated earlier, cost is always the driving factor in selecting or designing a system. The buzzwords of "modularity" and "flexibility" do not mean much when they are not backed up by a common systems design concept. Choosing a standard bus and off-the-shelf processor boards does provide modularity and flexibility, but only when the design goals are stated. For this IOS design, it is most important to maintain the functional decomposition while each function only has as much processing capability as needed. This concept is at the root of the design to production cost issue.

This system will allow the designer to implement normally expensive required instructor functions at a low cost. If the cost is still too high, the designer may swap the requirements and performance for a lower cost by maintaining the standard interfaces and localizing the hardware processing.

IOS PROGRAMMING

All instructional system application software is written in Ada. Although the C programming language has been standard in the graphics community, Ada will provide a more complete solution that addresses all aspects of the instructional system. The capabilities of Ada in addressing low level I/O in addition to enforcing good design practices make it the language of choice.

The concept of functionally partitioning the IOS tasks must be extended into the instructional system software. In other words, all menu structure routines should be handled as a group, as should instructor inputs, disk access requests, and database information interactions. This modularity and isolation of processing tasks supports future maintainability and expandability of the instructional system and assures that it can adapt to growth requirements. By clearly defining the interfaces between the tasks, upgrades or changes may be made to individual software modules as long as the interface is maintained.

Addressing and accessing the graphic capabilities of the IOS is as important as the instructional system application software itself. In order to have an immediate drawing capability with the delivered graphic system, the graphics system must provide its own graphic interface library. This library must remove all special graphic engine machine codes and routines from the programmer to let the programmer concentrate on application software, and not require him to be a graphics expert. The interface library should be based on an industry standard, such as GKS. Selection of this standard attempts to insure that if the graphics engine needs to be upgraded or replaced in the future, all the IOS software need not be rewritten.

CONCLUSIONS

The goal for this effort was to produce a single, core IOS design that could be replicated for most Honeywell trainer instructional systems. The design had to be cost-effective yet be flexible enough to adapt to many different and changing design requirements. The selection and use of commercial, off-the-shelf hardware and software, in the guise of Multibus II and GKS, will insure cost effectiveness through multiple source product availability. The selection of Ada as the design language further enhances the product by its ability to address a wide range of programming tasks in an efficient manner. Implementation of this design and further refinements can only improve its flexibility.

ABOUT THE AUTHOR

MR. PETER TUTKO is a staff engineer for the Flight Simulation Operation of Honeywell's Training and Control Systems Division. He received his BSEE from the University of Missouri-Rolla in 1983 and while there participated in a research study of a distributed microprocessor flight control system programmed in Ada. He worked as a hardware and systems design engineer for McDonnell Aircraft in their Flight Simulation Laboratory in the areas of advanced crewstation design, multi-processing simulator interface equipment, and network communications. His current assignments at Honeywell include computer architecture systems design and instructional and executive systems software.

A DIGITAL SIGNAL PROCESSING SOLUTION FOR SOUND SIMULATION

Brian P. Leger
McDonnell Aircraft Company
McDonnell Douglas Corporation
Department 251, Building 65
P.O. Box 516
Saint Louis, MO 63166

ABSTRACT

The design of sound simulators for aircraft and other vehicles has often presented a variety of problems in areas of integration, flexibility, maintenance and life cycle cost. Recent developments in digital signal processing (DSP) technology have provided a powerful and cost effective solution to these problems by way of a special device known as a single-chip digital signal processor. This technology allows fixed hardware to be highly flexible by using software algorithms to perform functions that would normally require analog oscillators, noise generators, filters and amplifiers. This approach eliminates recurring hardware design, simplifies integration, increases system reliability and provides better quality and control of sound parameters. This paper describes the features and advantages of a DSP-based sound simulator prototype that is capable of generating complex tone scenarios such as those found in avionic systems and other sounds such as those developed by a vehicle and its surrounding environment.

INTRODUCTION

In many training systems a sound simulator plays an important role in providing a trainee with a convincing and effective degree of realism. Over the years manufacturers have used a variety of methods for meeting sound simulation requirements including the use of actual recordings, design of unique sound generating circuitry, or, in more recent times, use of dynamically modified digital recordings. Interest in designing digital-based sound simulators has stemmed from demands for greater performance and reduction of life-cycle cost. The traditional analog-based approach has been very limiting with regard to meeting these cost/performance objectives because each type of training system has usually required a different sound simulator design. Furthermore, such systems have often required costly circuit changes to implement modifications of simulated sounds.

Digital signal processing (DSP) technology has grown significantly over the past few years and has forced many analog circuit designers to re-evaluate their options. Several companies have produced devices known as single-chip digital signal processors which can replace numerous analog circuits and are well suited to performing audio synthesis and processing. To demonstrate the capabilities of a single-chip DSP processor consider the case of simulating a missile launch from a fighter aircraft. The sound is typically simulated by generating random noise, filtering the noise to obtain the required center frequency and bandwidth and then amplitude modulating the signal with the proper attack and decay times. A single DSP processor has been programmed to perform this entire synthesis process. Similarly, a variety of other complex waveforms and tone sequences can be numerically defined and executed on a single-chip DSP processor. Sounds that are not required simultaneously can be grouped and executed by a single processor.

Using the approach outlined above, a DSP-based sound simulator prototype system has been developed which uses a software library to program fixed generic hardware for a specific training requirement. This paper discusses the following features and benefits of this prototype system:

- o A single type of circuit board is used as the system building block. The system can easily accommodate expansion and allows different types of trainers to be realized using the same hardware design.
- o Reliability is enhanced and periodic calibration requirements are eliminated by using digital components and implementing a built-in self-test.
- o 16-Bit digital-to-analog converters are used to supply wide dynamic range.
- o System can supply multiple analog output nodes for both monophonic and stereophonic sounds.
- o Numerical processing is used to create sounds. Enhances controllability and accuracy of sounds and allows quick turn-around time for system realization and modification.
- o Host integration is simplified by using a short command set and communicating via shared memory.

SYSTEM HARDWARE

The hardware architecture of this digital sound simulator is based on the use of multiple DSP processors to create a network of complex waveform generators. The network is formed using several copies of a single type of circuit board that allows the system to be expanded to meet the sound simulation requirements of a particular training system.

The DSP Circuit Board

The DSP circuit board is functionally and physically divided into the following three sections: A four-processor DSP array that executes sound synthesis programs, a communications interface that allows a host computer to communicate with the DSP processors through shared memory, and an analog section that provides two channels of digital-to-analog conversion and signal smoothing. The board's input/output ports allow two channels of synthesized audio to be digitally summed with signals on other boards, or the analog outputs

can be summed using an audio mixer. A simplified block diagram is shown in figure 1.

Proper operation of digital and analog circuitry is verified by a built-in self-test that executes at system reset or at the request of a host computer. Information supplied by the self-test allows a host computer to locate a faulty circuit board or identify specific failures on a given board. To minimize the impact of a component failure the circuit board makes no attempt to terminate its operation upon detection of a failure.

The hardware features identified above provide several advantages that have a direct impact on performance and life-cycle cost. Maximizing the use of digital components and implementing a self-test greatly increases reliability. The use of DSP processors provides extensive flexibility and eliminates recurring hardware design. Using shared memory simplifies system integration. A multi-board network comprised of a single type of circuit board minimizes the overhead required for support of the hardware.

The DSP Network

A network of signal processors is formed using several copies of the DSP circuit board in a single chassis. The number of boards required for a particular training requirement depends on the number of monophonic sounds, the number of stereophonic sounds, the number of sounds that are likely to occur simultaneously, and the number of stereophonic sounds that can be cross-coupled between processor pairs. A typical fighter aircraft may require between 8 and 10 of these DSP boards. The present addressing format allows for a maximum of 16 DSP boards per single chassis.

Once the required size of the network is determined, analog output nodes can be established based on the summing methods defined within the DSP software. The resulting analog signals may be connected to power amplifiers and other hardware such as an intercom system. Figure 2 shows how this DSP-based sound simulator can be implemented in a training system using

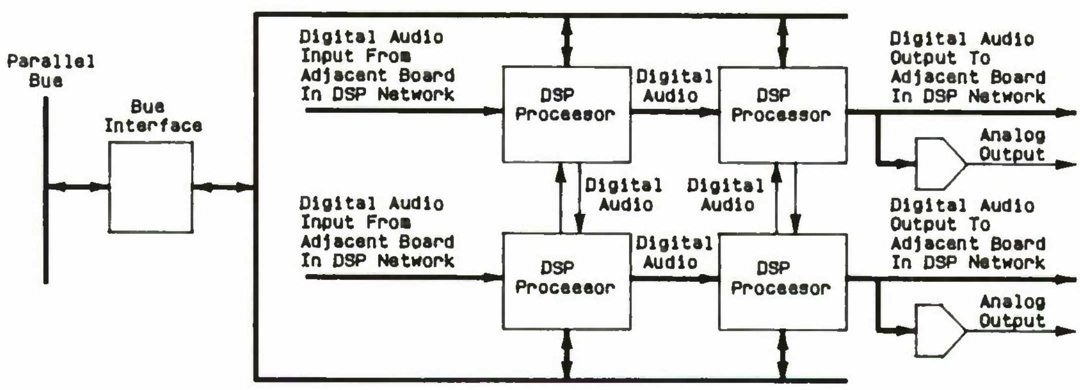


Figure 1. Simplified Block Diagram Of The DSP Circuit Board

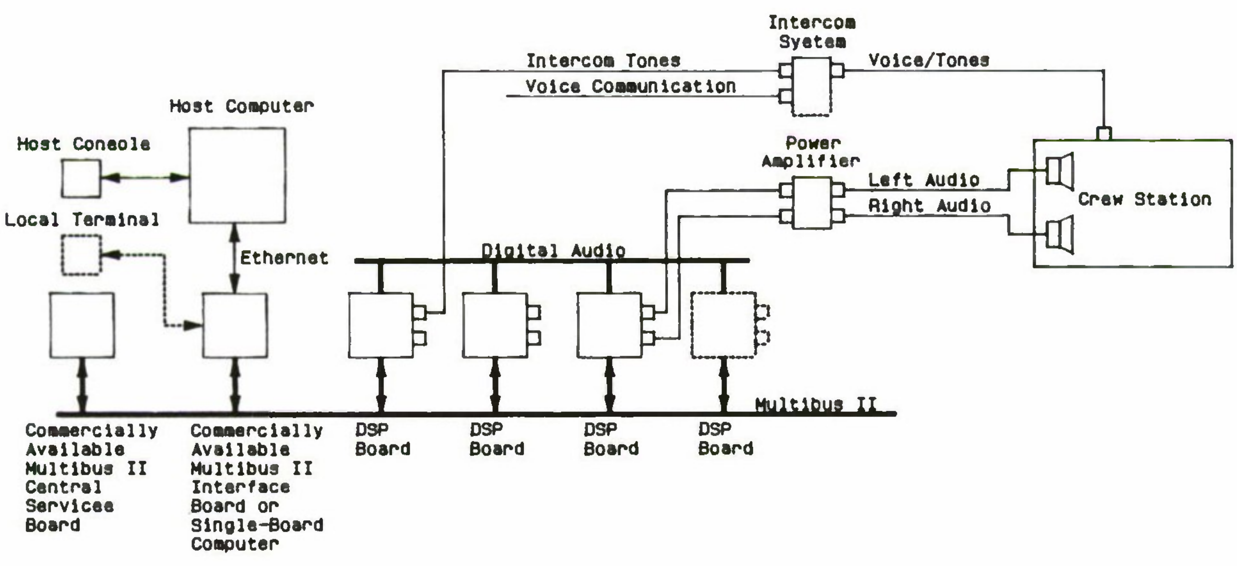


Figure 2. Implementation Of The DSP-Based Sound Simulator Using Ethernet and Multibus II

Ethernet® and Multibus® II.

Multibus is a registered trademark of Intel Corporation.
Ethernet is a registered trademark of Xerox Corporation.

SYSTEM SOFTWARE

Developing application-specific software for this system involves a five-part process of data acquisition, data analysis, sound simulator programming, host programming and system evaluation. This section describes the methodology used to perform these tasks.

Data Acquisition and Data Analysis

Obtaining real-world data typically involves using a recorder to capture the sounds on magnetic tape. The recordings are analyzed, and the unique features of each sound are identified. These procedures are common to most sound simulator designs and are not detailed in this paper.

Programming the DSP-Based Sound Simulator

Software support for this sound simulator includes a library of special purpose and general purpose algorithms that can be used to create sound synthesis programs. Waveform generating routines, filter routines and modulation algorithms are linked to create programs that can synthesize the time and frequency domain parameters that have been identified during analysis of audio recordings. Constants, coefficients and data tables are assigned by the programmer to specify frequencies, amplitudes, phase relationships, sweep rates, bandwidths, and the shapes of periodic signals and modulation waveforms. Using the available software support, programs can be developed with considerable savings in time and effort.

After the sound generating programs have been created, they are grouped as a single library of sounds that represent the requirements of a particular training system. Files within this library can be linked with the firmware used by the digital signal processors, or they can be transferred to a host computer and downloaded to the sound simulator's random-access memory.

Programming the Host Computer

Software for the host computer is developed using information obtained from analysis of real-world data. The host is responsible for providing control data such as on/off commands, specific data such as engine RPM and intensity, or one-time requests for sounds of finite duration. Although data may be sent at regular intervals, this sound simulator does not require refreshing unless the status of a sound or group of sounds actually needs to be changed.

The host is also responsible for downloading the sound generating programs to the sound simulator's random-access memory if the programs are not executed in the sound simulator's firmware. This can be done as part of system initialization, or, if the number of DSP boards is to be minimized, the host can download programs on-the-fly to instantly change the sounds emanating from the DSP processors. As an example, consider an aircraft training system which, among other sounds, requires simulation of runway rumble and gun fire. Since these sounds are not required simultaneously, they can be downloaded to the same DSP processor as required by the mission. If a user would prefer not to download programs from the host computer, the shared processor concept is still available by linking multiple programs into the firmware of a single DSP processor.

System Evaluation

The performance of each sound synthesis program can be compared with information obtained from analysis of real-word data. If discrepancies occur, the particular sound synthesis program(s) can be edited to adjust signal parameters. This evaluation task can be performed with an alternate host such as a single-board computer. The Multibus II interface board shown in figure 2 can be substituted with a single-board computer to allow local interaction with the sound simulator system. If desired, this single-board computer can remain as a permanent intermediate host that can serve both as a bus interface and as a local evaluation/diagnostic tool. Using support software, the sound generating programs can be exercised, evaluated and modified as required with rapid turn around time.

CONCLUSION

Digital signal processors have been used to develop a sound simulator that can be programmed to numerically synthesize the sounds required by a particular training system. Adopting a DSP approach has eliminated the need for recurring hardware design while preserving system flexibility and growth potential. Using numerical methods to create sounds allows a programmer to control virtually any sound parameter in a predictable and accurate manner. Life cycle cost is reduced by increasing system reliability and reducing the time and effort required for system implementation.

ABOUT THE AUTHOR

Mr. Brian Leger obtained his B.S. degree in electrical engineering from Western Michigan University in 1985. He is a laboratory Engineer with McDonnell Douglas Corporation and has been working in the Flight Simulation Department of McDonnell Aircraft Company for two years. Mr. Leger is a member of the Acoustics, Speech and Signal Processing Society of IEEE.

MPT&S GUIOANCE ANO CONTROL FOR WEAPON SYSTEM ACQUISITION *

Robert W. Stephenson, PhO
Plans and Operations Office (AFHRL/XO)
Air Force Human Resources Laboratory

Lt Col Frank C. Gentner, USAF
Manpower, Personnel, and Training Directorate (ASD/ALH)
Headquarters, Aeronautical Systems Oivision

ABSTRACT

The need for manpower, personnel, training, and safety (MPT&S) guidelines and constraints can originate at both the specific weapon system and aggregate system levels--whereas the typical Government acquisition team specializes only in information at the first (weapon system design) level. The amount of organizational support provided them is also not adequate to their task. In order to help integrate MPT&S factors during weapon system acquisitions, the Government needs: (a) enhanced analytic capabilities to analyze total system tradeoffs between man and machine in the performance, maintenance, and support of system tasks; (b) interactive communications with experts in system utilization policy and aggregate system constraints; (c) MPT&S-oriented incentive systems for Government, as well as for contractor personnel; and (d) a strong centralized headquarters advocate for MPT&S factors with the authority to establish policies and procedures for acceptable MPT&S guidance and control. Specific control guidance is also needed by Government acquisition teams and teams of contractor personnel. For this purpose, recent case studies of Government guidance and control were analyzed, and two lists of "do's" and "don'ts" were developed.

INTRODUCTION

It is difficult to get manpower, personnel, training, and safety (MPT&S) issues considered at an early stage during weapon system design. Government acquisition teams sometimes provide very little guidance about MPT&S issues because of uncertainty about the kind of guidance they should provide and reluctance to interfere with contractor operations. Contractors are almost forced to dictate MPT&S requirements under such circumstances. One of the reasons that this situation occurs is that the Government acquisition team does not have enough information to provide all the guidance that is needed.

On the assumption that experience is the best basis for facilitating MPT&S decisions, experience-based recommendations were collected from the literature as well as from experts in the field, and documented as recommendations in the paper that follows. Information alone, however, will not solve the problem. Organizational systems changes (improved analysis capabilities, improved communication and incentive systems, and new organizational structures) are also needed. The paper is thus intended for consideration by policy and decision makers as well as by Government and contractor personnel who work on the development of new weapon systems.

* The opinions expressed in the paper are those of the authors and do not necessarily reflect an official position of the Department of Defense or the U.S. Air Force.

THE ISSUES INVOLVEO IN GOVERNMENT GUIOANCE ANO CONSTRAINTS

In weapon system design, the most important priority is that the system perform as required. Other considerations, such as manpower, personnel, training, and safety (MPT&S) requirements are of secondary importance. There is a lot of merit in these priorities, since it would be very wasteful to develop a comprehensive MPT&S plan for each strawman version of a weapon system as it goes through the early concept exploration stages. One could conceivably develop 30 MPT&S plans, none of which would ever be used because the 3D strawman weapon systems for which the MPT&S plans were designed will never, in fact, be developed. It is only the approved weapon system design that will actually need MPT&S plans, and these plans will probably go through several iterations before they settle down.

There is, however, another side to this story. Assuming that the MPT&S plans are not taken seriously until the 31st iteration, problems are likely to occur. In the first place, the hardware system that "works" may not, in fact, perform as required if MPT&S factors are considered to be of secondary importance during the early stages. Assuming that the system does indeed meet expectations, the Government may find itself forced to accept a plan that is not realistic in terms of the available resources, or the Government could find that there is not enough time or money to develop the MPT&S systems (e.g., expensive simulators) that are needed before the system is scheduled to become operational. So, the Government, rightly or wrongly, encourages contractors to develop MPT&S plans early in the weapon system acquisition process (WSAP).

The amount of control that should be exercised is controversial. Too much Government control becomes excessive interference that can stifle the contractor's creativity or force the contractor to design a system in one particular way. It is always possible that the contractor might have used a different approach that could have saved the Government millions of dollars or been several times more effective if fewer restrictions had been imposed. At the other extreme, lack of Government constraints can become equally deplorable, since the contractor could waste millions of dollars designing something that is prohibitively expensive or cannot be used because the needed MPT&S systems are not available.

RECENT DEVELOPMENTS IN CONTROLS OVER MPT&S DECISIONS

Need for Early MPT&S Decisions. Several years ago, a number of advisory groups, including the General Accounting Office [1] and the Defense Science Board [2] urged the Government to consider MPT&S factors at an earlier point in the WSAP. In response , the military services made a number of efforts to change their procedures, but the initial results were not always fully satisfactory [3] [4]. Many problems can occur when MPT&S decisions are not made early in the WSAP, and the challenge of MPT&S integration was addressed in many different ways [5].

A good example of the need for early decisions is in the area of job aids. The ready availability of microcomputers makes it possible to modify MPT&S requirements extensively by using job aids and expert systems. Job aids can decrease the number of maintainers who are required, change high skill level jobs to low skill level jobs, decrease or change the training requirements, and convert unsafe conditions into safe ones. As pointed out by Lineberry [6], "...guidance with job aids should always be the choice, unless key factors contra-indicate, because job aids generally cost less to develop than instruction, are easier to revise when performance requirements change, reduce the time to achieve on-the-job performance, and are not subject to forgetting" (p 15). Booher [7] has provided a nine-step selection algorithm for identifying the most appropriate job-performance-aid/training combination. These decisions must be made early, since the job aids and expert systems could be built-in and become part of the equipment.

Control through Procedural Guidelines. All three services have developed procedural guidelines for controlling MPT&S decisions during the various stages of the WSAP. The Navy developed a system called HARDMAN [8] [9] (for Hardware and Manpower Integration), which was originally based upon some early Air Force work in this area [10] [11] [12] [13]. The Army has adopted similar techniques based upon an early version of the Navy system [14], and has recently expanded this approach to include even more areas of responsibility as part of a program called MANPRINT (for Manpower and Personnel Integration) [15]. Recent evaluations indicate that these procedural guidelines are working reasonably well [16] [17] [18], although there were a number of initial problems in getting the systems implemented.

Control through Data Item Descriptions (DIDs). Another approach to control is the use of standardized Data Item Descriptions (DIDs) which contain detailed descriptions of the kind of MPT&S plans that are to be provided by the contractor [19] [20] [21]. The DID needs vary from one stage of the WSAP to another. For example, the Navy [21] has one MPT concept DID, a separate MPT resource requirements DID, and a third MPT data report DID. Although revisions to these DIDs are not permitted, portions of the DIDs can be deleted to meet the needs of a specific weapon system. The advance thinking in these DIDs about what the Government should require at various WSAP checkpoints can be very useful, even when the original DID cannot be used.

SPECIFIC WEAPON AND AGGREGATE SYSTEMS GUIDANCE

Guidance and control are needed at the specific weapon system design and aggregate system levels.

At the specific weapon system design level, the major issues and concerns are ways of influencing the design of a weapon system and facilitating cost-effective performance of the personnel assigned to it. Qualitative and quantitative MPT&S requirements, key design characteristics for manning, job aiding, system maintenance, supporting job structures, and training--all of these must be evaluated with respect to optimum MPT&S performance for a specific weapon system. These analyses must be closely coordinated with human factors engineering specialists. Logistics support guidance is especially important, since it deals with how, where, and when the new weapon system will be operated, maintained, and supported. Examples of important logistics guidance decisions are: dispersed basing; maintenance concepts and the number of different levels of maintenance; operational temperatures, and the use of dedicated crew chiefs. Another important issue at the weapon systems design level is the need to establish an MPT&S baseline for determining the impact of proposed design changes.

Aggregate MPT&S systems combine information from several different weapon systems and jobs and examine MPT&S policy issues from an organizational unit, major command, and/or military department point of view. In aggregate systems, the major issues are the availability and affordability of manpower, personnel, training, and safety options in the context of the total force structure and all of the external demands that are made upon it. The important objectives are to avoid disconnects and unexpected consequences for MPT&S subsystems in future years [5]. Other issues at the aggregate systems level are cross utilization of information, reduced overhead requirements, and policy decisions to redesign or restructure occupational specialties. These analyses at the higher command level need to be continuously transmitted to specific product divisions for further planning and implementation.

WEAPON SYSTEM DESIGN GUIDANCE

MPT&S Guidance on the Way that Tasks are Assigned. One major MPT&S impact of weapon system design guidance is the way in which tasks and duties are assigned to the total (operator, maintenance, and support; civilian, military, and contractor support) man-machine system in order to make the weapon system operational. The constraints (e.g., operator maintainability, limits on mean time between failures) have important implications for the assignment of tasks to humans or machines, the cost effectiveness of the manned equipment system, the effectiveness of the multipurpose work group to which the individuals belong, and the extent to which that particular job assignment makes an individual more useful in future assignments.

One of the key issues in MPT&S system design is the amount and kind of specialization in jobs. On those occasions when a single weapon system will utilize all the time of the responsible personnel, the job design considerations are relatively straightforward [22]. What usually happens, however, is that many personnel are involved in each weapon system on a part-time basis. It is possible to design these part-time jobs such that personnel are specialized by function; to establish multifunction jobs in which personnel act as generalists; and/or to use computer software and job aids to minimize knowledge requirements.

Implications for Skill and Grade Progression Plans. The way in which tasks are assigned has important implications for skill and grade progression plans. Suppose that half the jobs in a particular occupational specialty involved assignments to generalist jobs and half the jobs involved assignments to specialist jobs. What would this do to career progression plans in that occupational specialty? Could technicians move back and forth between specialist and generalist assignments? Probably not, since the technicians would not be qualified for many of the tasks that they would be expected to perform in either case. The situation is complicated by the fact that overspecialized and underspecialized occupational specialties already exist. According to Edenfield [23], "Today's AF personnel specialty classification system, as it has evolved with advances in weapon system technology, has resulted in over-specialization/ job fragmentation in some disciplines and very broad-based, generic skills in other disciplines. These phenomena have resulted in a lack of work force stability and experience, inefficient use of manpower resources, poor job satisfaction and declining retention, and, possibly, an overstatement of manpower requirements" [23, p. vii]. These problems in Air Force Specialty Codes (AFSCs) are a direct result of the way in which tasks were assigned to personnel when weapon systems were designed in previous years.

The Impact of System Utilization Policy Constraints on Job Design. Several kinds of system utilization policy constraints could be imposed on the jobs that are performed by operations, maintenance, and support personnel. One possibility is to require that several functions be performed by the same person. For example, operators could be required to maintain their equipment to some degree (This is quite common in Army and Navy). It is also possible to require that the operators be assisted by job aids and computers. Another possibility is to impose limitations on the number of personnel that can be used when many different functions must be performed. This will usually force the contractor and/or the involved Government agencies to design generalist jobs that cut across traditional job specialties. Another option is to put limits on the amount of training that can be required or to put limits on the aptitude or skill levels of the incumbents. If the limits are restrictive, the contractor could be forced to design a system with lots of job aids, computer-assisted expert systems, "black boxes," etc. These tradeoffs should be analyzed early in the development cycle before resources are invested in options that will not be utilized.

MPT&S DECISIONS AT THE AGGREGATE SYSTEM LEVEL

Aggregate Data Bases and Information Systems. Each Service has a variety of limited purpose and aggregate information systems for manpower, personnel, training, and safety. The major function of these data bases and information systems is to ensure that there are no disconnects or unexpected consequences of decisions at the subsystem level among the organizations that are responsible for different parts of the MPT&S system. For example, if a new weapon system is going to require 1,000 additional fighter pilots and 10,000 maintenance and support personnel during a particular period of time, it is important that the manpower experts know that the slots are needed and distribute them to the right organizations, that the personnel experts set up assignment systems that will get people to the right places at the right times, that the training experts schedule the appropriate number of trainees into the appropriate training pipelines, and that the safety experts certify that the system is safe and make sure that the necessary safety regulations are issued and enforced in a timely fashion. Aggregate data bases and information systems are needed in order to do these things [24]--and they are needed years in advance. Aggregate data bases are also used by top-level decision makers when choosing among competing systems for inclusion in the future force structure.

The aggregate data bases could have important input-output relationships with job design and weapon system design decisions. These aggregate data bases provide: informed inputs regarding the total system consequences of specific weapon system designs; information about the MPT&S constraints that should be imposed upon weapon system design; and long range MPT&S planning inputs to aggregate system plans for future years.

Manpower, Aptitude and Skill Level Constraints. The most likely constraints to be imposed by decision makers at the aggregate level are constraints on the total number of personnel at each aptitude or skill level. As weapon systems have become more and more complex and technical, aptitude requirements--especially in the electronics specialties--have increased from year to year. Yet the labor market is not expected to change dramatically during the next

few years, and we will probably have approximately the same number (or less) of high aptitude people in 1995 as we have today. When skills are scarce, who will decide which weapon systems are really entitled to higher aptitude and/or skill level personnel, and which are not?

Each group of weapon system designers tends to think that their weapon system should be given priority over other weapon systems for the small number of military personnel who qualify for higher aptitude jobs. Yet we obviously cannot have job requirement profiles that do not correspond with the realities of the available military personnel populations from which those requirements must be met. It seems logical, then, to impose constraints on the system designers. For example, system designers can be prevented from requiring that their weapon systems be manned with nothing but engineering officers and E-7 technicians. If the long-range forecasters are expecting to have shortages in these categories--or, if the jobs that would prepare a person for E-7 skills do not exist (which prevents personnel from gaining the experience needed for higher level jobs)--the weapon system can be designed (using job aids, computer software, black box replacements, etc.) so that people with less skill, education, and aptitude can do what needs to be done. Moreover, we need to be certain that these forecasts will remain valid as systems go through development and are fielded for 10 years or more.

It is clear that the requirements for higher and higher aptitudes cannot continue indefinitely. The Army, which has historically been most affected by skill shortages, is taking an aggressive stand in this area with its MANPRINT program [15]. The other Services will be watching the Army's progress very carefully as it develops new systems and procedures for imposing manpower and skill level constraints on weapon system contractors.

Training Budget Constraints. It has become commonplace in recent years to require that the contractor provide crew maintenance and support training for a certain number of years after the new weapon system becomes operational. This has the effect of imposing training budget constraints that are likely to be tight if the original procurement was competitive. By establishing a financial cost if the contractor develops inadequate training systems, the Government hopes to receive better quality training systems in a more timely fashion.

RECENT STUDIES OF GOVERNMENT GUIDANCE AND CONTROL

Studies of Government guidance and control have been conducted in all three Services [25] [26] [27] [28] [29] [30]. Recommendations regarding guidance and control have also been provided as a result of conferences with industry [31]. The consensus is that the new MPT&S management systems (e.g., HARDMAN, MANPRINT) are being used and are having a beneficial effect.

Most of the problems that have occurred can be attributed to less-than-adequate, biased, or excessive control by the Government. The following statements summarize expert opinions regarding the "direct causes" of the human factors and MPT problems that have occurred .

UNDERCONTROL

There was ambiguity and/or lack of precision in describing required system objectives.

System description was incomplete.

Task and skill analyses and man-machine tradeoff studies were not required early enough to affect basic systems parameters.

Many of the proposed MPT&S measures could not be verified or enforced.

There was laxity in following up and verifying human factors and MPT&S supportability goals.

Test and evaluation plans did not emphasize maintenance support requirements in operational environments.

Inadequate guidance and unmeasurable criteria were contained in requirements documents.

MPT&S decision points and evaluations for new systems were programmed without adequate test or evaluation.

Design requirements for training equipment were very general and incomplete.

No penalties were established for failure to perform MPT&S planning.

OVERCONTROL

Some systems requirements were specified exactly when they should have been determined by tradeoff analysis studies.

STATUS QUO APPROACH

MPT&S approaches that had worked for previous systems were accepted uncritically without proper examination of the unique circumstances of the system currently under development.

Personnel characteristics of previous systems were assumed to be valid for new systems, without adequate test or evaluation.

Maintenance requirements were assumed to be met with routine and standard maintenance procedures when other options should have been explored.

Manning was by policy rather than by requirements.

HARDWARE BIAS

There was a tendency to overlook personnel-oriented performance measures and man-machine tradeoff studies in favor of equipment development.

In performance specifications there was too much concentration on hardware rather than man-machine performance.

There was a tendency to overlook human performance measures in favor of hardware-oriented performance measures.

The general attitude was, "Let's worry about the equipment first; we can always get the people later."

RECDMMENDED GUIDANCE FDR WEAPDN SYSTEM DESIGN

Based upon our analysis of the case studies reported in the literature and conversations with experts in the field, a slightly different approach seems to be needed at each stage of the WSAP (see Table 1).

Pre-Concept. During the pre-concept phase, the Government needs some way of specifying constraints without telling the contractor how to design the weapon system. These constraints are required because of the circumstances under which the weapon system would be used. For example, limitations on maintenance manpower could be created because of dispersed basing requirements. Even though these constraints are imposed by system utilization policies, it is still possible to give the contractor enough freedom to come up with a range of personnel mixes in support of the type of weapon system desired. The contractor can be required to conduct broad-brushed total system trade studies before recommendations are made regarding the design of specific MPT&S subsystems.

Contractors do not want to be perceived as "non-responsive." They will usually give the Government what it says is wanted, unless there are strong reasons to do otherwise. So the Government acquisition team must be very careful about what the Government "says" is wanted. Dn the other hand, the performance of work costs money--and the contractors will not perform work that is "implied" or "seems to be" a logical requirement unless there is an explicit requirement that they do so. This is especially true of tradeoff and sensitivity studies for MPT&S alternatives. It is important that the requirement for such studies be explicitly stated in the Request for Proposal (RFP) when it is issued. It is also important that the tradeoff-thinking implicit in such a requirement not be negated by other requirements in the RFP. The Government should not ask the contractor to plan and conduct manning tradeoff studies, for example, while simultaneously requiring that the weapon system be operated by a two-person crew. Another important point to remember here is that good MPT&S systems will not be free. If the Government wants high quality MPT&S systems, it must pay for it.

Too often, the pre-concept constraints are decided upon by contacting the headquarters organizations with responsibility for each relevant area of expertise, and arriving at a consensus. The time available for studying these issues at these headquarters organizations is rarely adequate for a comprehensive study of constraint alternatives. The headquarters organizations cannot always be as future-oriented as they should be, since they are very busy trying to keep track of the status quo; nor do they usually have available the kind of long-range oriented analytic capabilities that are needed; and the aggregate data bases that are needed to justify constraints are not always available.

TABLE 1. Recommended Approach for Weapon System Design

WSAP PHASE	PRESENT APPROACH TO MPT&S REQUIREMENTS	RECOMMENDED APPROACH TO MPT&S REQUIREMENTS
Pre-Concept	Consensus of responsible organizations that are primarily responsible for the status quo	Creative analytic studies of MPT&S constraint alternatives, goals, and issues
Concept Evaluation	Engineering-oriented trade studies	Total-system-oriented trade studies (including MPT&S alternatives) for both operators and maintainers
Demonstration-Validation	Budget is usually adequate only for engineering system improvements	Adequate budgets for total system improvements and alternate system analysis
Full-Scale Development	Quick fixes for inadequate or underdeveloped MPT&S systems	Evolutionary changes only, since MPT&S system needs are already anticipated
Production & Deployment	Gradual evolution of MPT&S system improvements	Minor changes only, since MPT&S system needs are already anticipated

Concept Evaluation. The typical concept evaluation trade study at the present time is engineering-oriented. What is needed instead are total-system-oriented MPT&S trade studies (including both operator, maintainer, and support personnel) in which man-machine tradeoffs are considered. These tradeoff studies cannot be permitted to become "pencil-whipping" exercises in which evaluations are based upon superficial analyses of alternatives that are not really competitive. In-house Government expertise and independent quality control checks are needed in order to make certain that the concept evaluation trade studies are well conducted and taken seriously.

Demonstration-Validation. A common conclusion after competitive procurements are awarded is that the demonstration-validation budgets are adequate only for engineering system improvements; MPT&S plans (and possibly logistics and maintenance plans as well) are often curtailed because the engineering budgets were underestimated. It may be hard for the Government to do this at times, but someone needs to step in, evaluate the plans, recognize that the budget is inadequate, and take whatever steps are needed to ensure either that the budget is increased or that the work plans are modified to redefine the system. This may be difficult to do when the company has a fixed price contract and there are already cost overruns and schedule slippages--but someone must do it if MPT&S factors are to be given the weight that they deserve.

Full-Scale Development. The typical approach during full-scale development is that of quick fixes to resolve MPT&S oversights. Evolutionary changes are to be expected; however, very few quick fixes should be needed if the MPT&S requirements are properly anticipated (and given realistic budgets) during earlier WSAP phases. The MPT&S efforts during full-scale development should be devoted to refining the MPT products developed earlier (numbers, skill levels, tasks, and training analyses). Test/evaluation and validation of these MPT&S projections need to be programmed. In addition, evaluation of training development, training media and materials (formative and summative) will be a major activity.

Production and Deployment. When the new weapon system is actually deployed, there may still be a need for some quick fixes, but one thing is certain: If the MPT&S requirements were not understood before, they are about to become understood in a hurry. For obvious reasons, operational personnel are active proponents of improvements that would make the system more effective and cost-efficient. MPT&S systems will consequently improve during production and deployment in an evolutionary way as fast as circumstances will permit. There is nothing wrong with this process, and nothing wrong with the importance attributed to MPT&S factors (at long last). Ideally, however, the MPT&S needs would have been adequately anticipated in previous stages, and little change should be needed during the production and deployment stages.

CONSTRAINTS ON THE MPT&S PROCESS

Weapon System Design Constraints. Almost everyone is willing to agree that MPT&S utilization policy and task assignment constraints should be developed and imposed during the pre-concept and concept evaluation phases. Unfortunately, the Government personnel responsible for developing a weapon system usually do not have a clear-cut idea as to what these constraints should be. System utilization policy, skill level, and task assignment constraints are hard to specify when the exact nature of the equipment is unknown and the equipment developers and the MPT&S experts are in separate organizations. They are, however, no more difficult to specify than the equipment options under consideration. The most important impact of system utilization constraints is on the assigment of functions to man or machine. New and improved total system analysis techniques are needed to evaluate the pros and cons of assigning tasks to human personnel, to machines, to human personnel equipped with job aids, to specialists, to generalists, etc. Logistics system constraints are especially important. Examples are: dispersed base locations; maintenance levels; dedicated crew chiefs; requirements for operator maintainability; limitations on the number of maintenance personnel available to support a system; and requirements for the consideration of machine-assisted alternatives that would limit crew size. Clearly, the MPT&S developers need to work closely with human factors engineers in order to deal with these constraints.

Aggregate System Constraints. Aggregate system constraints usually derive from the projected availability of personnel at particular skill levels, the feasibility of establishing new occupational specialties to support a particular weapon system, acceptable training times, etc. It is important that this guidance be provided in a flexible format that permits tradeoff studies. It is also possible to be more directive. One Army general recently directed, for example, that the Army establish a Design for Discard (DFD) program. The emphasis in DFD was to be "innovative design to reduce the cost of discard" rather than repair cost analysis or classical engineering approaches [32]. The general decided on this approach because of manpower projections that fewer people with higher skills would be available when needed and excessive "tooth to tail" (i.e. combat to logistics support) ratios. Ideally, however, aggregate data bases would be used to provide guidance without ruling out viable alternatives when new weapon systems are designed.

THE NEED FOR ENHANCED ANALYTIC CAPABILITIES WITHIN GOVERNMENT

It is easy to tell Government representatives that they should provide more information about system constraints. It is not easy to tell them how to do it. Nor is it really clear who should conduct the quality control checks and provide the weapon system designers with the kind of guidance that is needed.

Many analytic procedures already exist to justify constraints at the weapon system design level. This is not as true of constraints that logically originate at the aggregate system level. Neither the Government contract monitors nor the weapon system contractors are likely to have the expertise that is needed to say what these constraints should be. They rarely have access to long-range forecasts and long-range plans; they rarely have the "big picture"; and they are not supposed to establish policy.

Each military department has "studies and analysis" groups that conduct constraint-oriented studies of the type that is needed--but they are rarely available to study specific weapon system constraints on short notice. New data bases, analytic methods, and study groups seem to be needed in order to help expedite this process. Important tools and guidelines needed by MPT&S study groups are: ways of stating MPT&S requirements in terms of criteria that can easily be measured; ways of dealing with the interfaces between subsystem data bases; and ways of forecasting the impact of weapon system design decisions on MPT&S criteria at early stages during the design process. The data bases and methods should be a computerized system that would include systems characteristics, logistics, MPT&S factors, warfighting capability, and costs. The new data bases and analytic methods should assist and interact with the MPT&S analyst in a "decision support" mode [33], and help get his or her inputs considered during relevant facets of the weapon system design process, hopefully including an interface with the computer assisted design (CAD) process. The system should be capable of simulating wartime scenarios given various inputs (reliability rates, numbers of people, etc.). The system should also permit various levels of analysis--top level as well as more specific options.

The new guidelines and decision aids are needed to make it easier to model a new system in terms of its complexity, types of components, and MPT&S requirements. Analytic methods that are capable of evaluating tradeoff decision options and identifying the best options for further exploration are also needed. Given these decision aids and data bases, early budgeting and MPT&S requirements could be based on historical records and growth/cost curves. These early MPT&S estimates could then be refined (possibly using computer-assisted update systems) as more specifics are learned during the design and developmental processes.

Unified data bases [34] seem to be logical prerequisites for these MPT&S decision support systems--but a lot of work still needs to be done, in spite of the many procedural guidelines that already exist. We are a long way from the system described in the preceding paragraphs.

THE NEED FOR INCENTIVES

To make the MPT&S system work, it is possible to use the same incentives approach that was used with the Air Force Reliability and Maintainability (R&M 2000) program that was signed into action on 1 Feb 1985 [35] [36]. This would require: clear statements of MPT&S needs in official requirements documents throughout the entire WSAP; quantitatively stated requirements to select MPT&S systems that are systems-effective and cost-efficient; improved source selection procedures that would give more weight to the past MPT&S record of the companies that are being evaluated; the documentation of "lessons learned" regarding MPT&S system tradeoffs and their dissemination to all involved contractor organizations and Government agencies; contract incentives and warranties that would guarantee satisfactory MPT&S systems for a given number of years after the system becomes operational; contract evaluation points that are timed to correspond with the satisfactory development of MPT&S systems; specific requirements for timeliness and ready accessibility of needed MPT&S products; specific requirements for field evaluations of MPT&S systems before the implementation phase is reached; and a DOD-wide coordinating group that would ensure that new ideas for improved MPT&S systems are put to work in an expeditious fashion.

A similar set of incentives is needed to avoid disconnects and unexpected consequences within Government organizations. For the contractors, money is the best incentive. For Government MPT&S managers, the best incentive is to provide prompt cost-effectiveness feedback to the managers of those who make the planning decisions. Qualified evaluators and enhanced study analysis capabilities are needed to provide the kind of feedback that is needed. General officer support is needed to make certain that the evaluations are taken seriously.

THE NEED FOR CENTRALIZED HEADQUARTERS COORDINATION GROUPS

Although all three Services have established headquarters focal points for MPT&S systems, the authority and the resources allocated to these headquarters groups have not always been adequate. The current headquarters staff groups in the Air Force do not have enough influence or resources to insist upon or support analytic studies of system utilization policies and aggregate system constraints, for example.

Since all three Services are working this problem area using similar policies and procedures, it may be desirable to set up a DOD-wide Headquarters Coordination Group for MPT&S systems. An organization along these lines already exists in the training area--the Training and Performance Data Center (TPDC) [37]. It is possible that TPDC could be modified to give it a broader perspective so that it could accept more responsibilities in the MPT&S area.

Even if the TPDC role is broadened, however, a strong headquarters focal point for MPT&S factors is needed within each military department. It is very important that headquarters coordinators have the authority to direct that MPT&S policies and aggregate systems guidance be followed by lower echelons. The need for such a group in the Air Force was recognized in the recent Akman Associates report [24] on the design of Air Force systems for Readiness Achieved through Manpower Personnel, Requisite Training, and Safety (RAMPARTS). An important proposal in their report was that a strong, centralized office be established within the Air Force.

One of the most important objectives for new organizational structures is improved communications between weapon systems designers, data base designers, and experts at the aggregate systems level. People need to talk to people--to ask questions, get expert advice, and let the experts know how well their recommendations worked out; and these communications need to take place quickly. Designing communication systems of this type is an important challenge for those who would establish new organizations to facilitate the MPT&S planning process.

"DO'S AND DON'TS"

We prepared two lists of "Do's" and "Don'ts": one for Government acquisition teams (Table 2) and one for teams of contractor personnel (Table 3). We then sent preliminary drafts of Tables 2 and 3 for review by approximately 20 experts in the field. As a result of their comments, some additional items were added to the lists, and some of the original items were revised or deleted. The editorial decisions are ours, however; so the two lists do not represent a consensus.

Table 2

"Do's" and "Don'ts" for Government Acquisition Teams

MOST RELEVANT WSAP PHASE	DO	DON'T
PRE-CONCEPT	Specify what the weapon system must do within constraints without telling the contractor how to design the MPT&S systems for it.	Don't tell the contractor how to design the MPT&S systems for a specific weapon system.
	Focus on total system performance, including all of the human performance aspects.	Don't focus on operational system performance requirements and neglect MPT&S requirements.
	Specify MPT&S system objectives completely and unambiguously.	Don't use ambiguous or incomplete descriptions of MPT&S system objectives.
	State the implications of future military demographics for MPT&S system design in the context of total force commitments when the system will be fielded and continue to operate.	Don't design the system for present populations or populations that have infinite skills and abilities; and don't underestimate the intelligence, and desire to be proud of job accomplishment, of future military personnel.
	Provide supporting literature and documentation to the contractor in a timely manner.	Don't force the contractor to waste resources on unnecessary red tape for getting access to needed information.
	Consider requiring that the contractor use validated MPT&S analysis methods (e.g., the Logistics Composite Model (LCOM)) for manpower modeling.	Don't allow the contractor free reign in selecting non-standard or unvalidated MPT&S analysis techniques.
	Tell the contractor what the most important problems are on comparable systems and request that the prime contractor design these problems out of the new system; develop MPT&S lessons learned and provide them to the contractor.	Don't assume that the contractor will understand MPT&S problems found in other weapon systems and design the system accordingly without special guidance on your part.
CONCEPT EXPLORATION	Be sure that requirements for MPT&S sensitivity and tradeoff analysis studies are explicitly stated in the initial request for proposals.	Don't assume that logically "implied" tasks will be performed when there is no explicit requirement that the contractor perform them.
	State MPT&S constraints, goals, and issues to be examined in such a way that a reasonable range of tradeoff decisions is permitted.	Don't impose constraints that eliminate viable MPT&S tradeoff decisions.

Weigh MPT&S factors heavily in the procurement and hold the winning contractor to promised performance.	Dont' permit MPT&S factors to be assigned a low priority when budget and authority decisions are made.
Evaluate contractors on past as well as promised performance. Let them know that their track records in MPT&S are to be considered in future procurements.	Don't permit past performance in MPT&S system to be overlooked when procurement decisions are made.
Include MPT&S experts on source selection review panels and on design review teams. Use outside consultants where government personnel are weak.	Don't assemble selection panels or design review teams that do not contain the kind of MPT&S systems expertise that is needed.
When developing the human factors plan, ensure that attention is paid to the maintainer as well as the operator, since lack of system availability due to poor maintainability can lose the war.	Don't concentrate most human factors efforts on the operator, leaving maintenance human factoring for engineering change proposals after the system is fielded. This can cause expensive fixes, critical delays in maintenance turnaround, and unrealistic demands on the resource requirements for future MPT&S systems.
Consider effects of wartime environmental factors on MPT&S (e.g. chemical-biological, aircraft battle damage).	Don't base supportability analyses only on peacetime requirements.
Prioritize the MPT&S analyses that are needed; be prepared to justify their value and show how you will use the information.	Don't request nice-to-have data that no pressing need requires or request data that will be received too late to be used effectively.
Enforce MPT&S, human factors, reliability and maintainability (R&M), and other supportability design criteria from MIL STDs in your program.	Don't waive supportability MIL STD requirements applicable to your program, since they could affect life cycle costs, have severe MPT consequences, and/or affect force readiness.
Permit and encourage tradeoffs between M, P, T, and S factors on a variety of cost, performance, and other supportability criteria.	Don't assume that one kind of tradeoff study is all that is needed.
Tell the offerer how MPT&S and human factors engineering (HFE) factors will be evaluated and the relative weight that will be given to supportability.	Don't fail to provide information about the weight assigned to MPT&S/HFE evaluation factors.
Have the prime contractor model the consequences for MPT&S supportability and indicate design changes that could improve it.	Don't overloook modeling of MPT&S supportability issues as a technique for analyzing interactions within the system.
Develop objective source selection criteria that will differentiate among proposals that have MPT&S supportable systems and those that may have problems.	Don't use subjective or incomplete proposal selection criteria that will permit MPT&S problems to go unnoticed.

	Require that important training environment conditions be considered in the design of MPT&S systems.	Don't permit contractors to design training systems that can be used only under uncontaminated conditions at large bases and/or at well equipped training centers.
	Be sure that each MPT&S task has a product, and that the products are required deliverables.	Don't assume that mere recitation of the goals of the weapon system means that the contractor will spend adequate time and money on MPT&S tasks.
	Use a DOD Form 1423 to require delivery of MPT&S products to the Government.	Don't assume that if the Statement of Work (SOW) contains MPT&S tasks, the products will automatically show up on your desk.
	Be sure that the MPT&S performance requirements are clearly defined, quantifiable, and testable.	Don't specify MPT&S objectives in terms of criteria that cannot be measured.
	Require that the contractor consider alternative MPT&S concepts that would control, avoid, or reduce safety and health hazard risks.	Don't accept safety and health hazard risks because they are considered basic to a particular technical concept.
DEMONSTRATION VALIDATION	Make sure that plans for MPT&S systems remain up to date as plans for the weapon system evolve.	Don't allow MPT&S plans to lose concurrency with plans for engineering design changes.
	Schedule MPT&S contract reviews concurrent with other system evaluations.	Don't permit MPT&S factors to be overlooked during engineering system evaluations.
	Require cost-effectiveness and cost-benefit evaluations of the most important MPT&S alternatives.	Don't accept MPT&S justifications of important systems that do not contain adequate cost-effectiveness and cost-benefit evaluations.
	Subject the prime contractor's MPT&S models to close scrutiny by asking Government support contractors to conduct an independent evaluation using an independent systems manpower model (e.g. LCOM). (NOTE: Independent life-cycle cost (LCC) analysis is required by legislation, and manpower is a major cost element of LCC).	Don't assume that the prime contractor will understand the full MPT&S implications of the developing system.
	Select appropriate Logistics Systems Analysis DID requirements and specify timely delivery in appropriate formats of all data needed for MPT&S.	Don't assume that all participants are planning to jointly use common data items; if coordinated data are not required, some data needed for later decisions may not be placed on contract, or you may pay for essentially the same data twice.
	Require analytic support of MPT&S requirements for new system proposals.	Don't assume that past practices are an acceptable model for new systems without analytic support.
	Make sure that the funding priority of MPT&S factors is protected during changes and perturbations in the WSAP.	Don't permit MPT&S budgets to be cut in order to provide additional funds for other purposes.

FULL SCALE DEVELOPMENT	Require MPT&S managers to sign off on all design drawings and design changes.	Don't assume that engineering changes will have little or no effect on MPT&S performance.
	Require that the contractor have qualified MPT&S personnel and adequate MPT&S budgets.	Don't assume that qualified personnel and adequate budgets will be provided when needed.
	Qualitatively and quantitatively verify that MPT&S factors have been adequately considered.	Don't approve plans without qualitative and quantitative verification of MPT&S systems by the Government.
	Consider beginning training development contracting early enough to allow development and delivery of training systems concurrently with first weapon system delivery.	Don't assume that training development can play "catch up" to system production using compressed scheduling without allowing for prohibitive costs. It usually ends up as expensive interim contractor support with very little, if any, scheduled relief.
	Require that the contractor consider meaningful alternatives to the proposed MPT&S plans.	Don't accept the recommended MPT&S plans without requiring consideration of alternatives that are viable and competitive.
	Test and evaluate the adequacy of MPT&S numbers and skill requirements.	Don't assume that the contractor's estimates will be acceptable without some kind of verification.
	Ensure that tests and evaluation is conducted using realistic tests of MPT&S support systems using "average" personnel of the type expected to operate, maintain, and support the system.	Don't allow tests to be conducted using only "superhuman" personnel who can make anything work.
	Continue to monitor the impact of MPT&S criteria in the design of the weapon system by participating in regularly scheduled conferences and reviews.	Don't assume that your job is done just because the system has reached the full-scale development stage without any major problems.
PRODUCTION AND DEPLOYMENT	Require that adequate MPT&S resources be allocated for cost and risk reduction studies during redesign and retrofit.	Don't expect cost and risk reduction studies to be conducted without adequate resources to support them
	Be alert to the possible impact of procurement changes (e.g., an accelerated schedule) on the design of MPT&S systems.	Don't assume that changes in procurement will have negligible effects on MPT&S plans.
	Document MPT&S lessons learned so others can benefit from this experience.	Don't assume that MPT&S problem histories won't be repeated.

Table 3. "Do's and Don'ts" for Teams of Contractors

MOST RELEVANT WSAP PHASE	DO	DON'T
PRE-CONCEPT	Start MPT&S planning early and focus on performance of the total system (operator, maintainer and support).	Don't focus on hardware and defer consideration of operator, maintainer, and support functions until later.
	Conduct task and skill analyses early enough to let your findings influence the design.	Don't defer task and skill analyses until hardware requirements are firmly established.
	Consider the costs and benefits of substituting technology for labor	Don't assume that traditional assignments of tasks to labor will be cost effective for the new system.
CONCEPT EVALUATION	Address MPT&S factors for all system components, conditions, and major scenarios.	Don't limit MPT&S studies for budget reasons.
	Plan MPT&S support for the entire life of the system.	Don't limit your MPT&S plans to the first few years.
	Allocate sufficient protected budget and authority to MPT&S contractor personnel/organizations so that they have the means to perform.	Don't undermine efforts by glossing over or ignoring MPT&S problems that will affect schedule or budget.
	Provide accurate estimates of the total system manpower, skill, and aptitude requirements for all circumstances under which system operation is expected.	Don't underestimate or overestimate the total system manpower, skill, and aptitude requirements.
	Plan for concurrent and coordinated development of training systems (operator, maintenance, support).	Don't postpone training planning until later acquisition phases.
	Consider the occupational specialty and career field implications of the system being developed (overspecialization, aptitude requirements, new skills).	Don't assume that existing military occupational structures and resources will provide the kind of personnel that are needed.
	Make sure that the individual MPT&S area managers continuously coordinate developments in their areas with the other area managers.	Don't assume that a development or change in one "people" area will have no effect on other "people" areas.
DEMONSTRATION VALIDATION	Use human performance measures and standards to help influence and evaluate system performance and supportability, since they will affect performance.	Don't use engineering performance studies alone to evaluate system performance.

	Develop plans for dealing with potential safety and health hazard impacts in all environments in which the system will be used.	Don't limit safety and health hazard studies to the most likely environments in order to reduce costs.
	Plan for methods to reduce "skill decay" (resulting from lack of frequent practice) through cost effective changes in training system design.	Don't focus on initial skills acquisition without giving adequate attention to problems resulting from lack of frequent practice.
	Conduct human-machine tradeoff studies that optimize maintenance and support of the total system as well as consider machine tradeoffs.	Don't limit human-machine tradeoff studies to operators and equipment.
	Conduct human-machine tradeoff studies early enough to influence the design of the equipment.	Don't defer human-machine tradeoff studies until equipment characteristics are relatively well established.
	Require MPT&S managers to sign off on all design drawings and design changes.	Don't assume that engineering changes will have little or no effect upon human performance.
FULL SCALE DEVELOPMENT	Analyze and develop total system training requirements (equipment, spares, technical manuals) for all personnel.	Don't assume that technical training for operators alone will fully support the system.
	Evaluate the wartime and peacetime system impacts of proposed changes in weapon system design to ensure supportability from an MPT&S point of view.	Don't evaluate proposed systems changes without considering both wartime and peacetime impacts.
PRODUCTION AND DEPLOYMENT	Allow adequate time and resources for evaluating MPT&S systems before operational use is required.	Don't delay development of MPT&S systems so long that there is not enough time or money to support adequate testing before operational use.
	Make sure that training equipment is available when new systems are fielded.	Don't field new systems without needed training equipment.

CONCLUSIONS

Existing Government guidelines and constraints for those responsible for MPT&S factors in Government acquisition teams are not working well. There are many instances of: undercontrol; overcontrol; too much use of a status quo approach; and a strong hardware bias. Providing experience-based guidance to Government and industrial personnel will go a long way towards improving the situation, but it is not enough.

Satisfactory guidance and control are not likely to be forthcoming unless the following steps are taken: the development of enhanced analytic capabilities that can analyze system utilization policies and make tradeoffs between man and machine in performance of system tasks; the establishment of interactive communication channels between experts in weapon system design and aggregate system constraints; the establishment of incentive systems that will reward both Government and contractor personnel for giving greater priority to MPT&S factors in weapon system design; and the establishment of a strong, directive headquarters group that can act as an advocate of total-system-oriented MPT&S plans within each military department.

REFERENCES

[1] General Accounting Office (1981, January). Effectiveness of U.S. Forces can be increased through improved systems design (Rep PSAD-81-17). Washington D.C.: Author (AD A114237).

[2] Defense Science Board (1982). Report of the Summer Study Panel on Training and Training Technology. Washington D.C.: Office of the Undersecretary for Research and Engineering.

[3] O'Connor, F.E., Fairall, R.L., and Birdseye, E.H. (1984, January). Determination of Manpower, Personnel, and Training Requirements: A Synthesis of Case Study Findings (Summary Report) (Research Note 84-34). Alexandria VA: U.S. Army Research Institute for the Behavorial and Social Sciences (AD A137 708).

[4] General Accounting Office (1985, September). The Army Can Better Integrate Manpower, Personnel, and Training into the Weapon Systems Acquisition Process (GAO/NSIAD-85-1540). Washington D.C.: Author.

[5] Stephenson, R.W. & Ulrich, T.E. (1986, November). The Challenge of MPT Integration. Proceedings, Interservice/Industry Training Equipment Systems Conference. Washington D.C.: National Security Industrial Association, pp 116-125.

[6] Lineberry, C.S., Jr. (1976). When to develop aids for on-the-job use and when to provide instruction. In T. Rowan and J.J. Genevese (Eds.), Second Annual Job Information Presentation Conference. Buffalo, NY: The Society for Applied Learning Technology.

[7] Booher, H.R. (1978). Job Performance Aid Selection Algorithm: Development and Application (NPRDC-TN-79-1, AD F630 037). San Diego, CA: Navy Personnel Research and Development Center.

[8] HARDMAN Development Office (1985, May). HARDMAN Methodology: Equipment/System/Subsystem (HARDMAN Publication No 84-01). Washington D.C.: Chief of Naval Operations (OP-111).

[9] HARDMAN Development Office (1985, July). The Navy Program Manager's Guide to Early MPT Planning (OPNAV P-111-13-85). Philadelphia, PA: Navy Publications and Forms Center.

[10] Goclowski, J.C., King, G.F., Ronco, P.G. & Askren, W.B. (1978, March). Integration and Application of Human Resources Technologies in Weapon System Design: Coordination of Five Human Resource Technologies (AFHRL-TR-78-6 (I), AD-A053680). Wright-Patterson AFB, OH: Advanced Systems Division, Air Force Human Resources Laboratory.

[11] Goclowski, J.C., King, G.F., Ronco, P.G., & Askren, W.B. (1978, March). Integration and Application of Human Resources Technologies in Weapon System Design: Coordination of Five Human Resource Technologies (AFHRL-TR-78-6 (II), AD-A053681). Wright-Patterson AFB, OH: Advanced Systems Division, Air Force Human Resources Laboratory.

[12] Goclowski, J.C., King, G.F., & Askren, W.B. (1978, May). Integration and Application of Human Resource Technologies in Weapon System Design: Consolidated Data Base Functional Specification (AFHRL-TR-78-6 (III), AD-A059298). Wright-Patterson AFB, OH: Advanced Systems Division, Air Force Human Resources Laboratory.

[13] Preidis, R.J. (1984, September). Computerized Technology Package--ASSET: Test and Evaluation (AFHRL-TR-84-6, ADA146486). Wright-Patterson AFB, OH: Logistics and Human Factors Division, Air Force Human Resources Laboratory.

[14] Mannie, T.E. & Risser, D.T. (1984, February). Estimating Manpower, Personnel, and Training Requirements Early in the Weapon System Acquisition Process: An Application of the HARDMAN Methodology to the Army's Division Support Weapon System (Tech Report 616). Arlington, VA: U.S. Army Research Institute for the Behavioral and Social Sciences.

[15] Department of the Army (1986). Army Regulation 602-02. Manpower and Personnel Integration (MANPRINT).

[16] Council, G.S. (May 1985). A Review of the Navy HARDMAN Pilot Implementation Program. March 1983 - May 1985. Internal HARDMAN Office report. Arlington VA: OPNAV HARDMAN Implementation Office.

[17] Zimmerman, W., Butler, R., Gray, V., & Rosenberg, L. (1984, August). Evaluation of the Hardman (Hardware vs. Manpower) Comparability Methodology (Tech Report 646). Alexandria, VA: U.S. Army Research Institute for the Behavorial and Social Sciences.

[18] Smith, J. V. (1986, May). Technical Review and Analysis of the HARDMAN Comparability Analysis Methodology. MRSA LSAT 86-01, Lexington, KY: USAMC Material Readiness Support Activity (MRSA).

[19] Cherry, W. P., Promisel, D. M., and Miles, J. L. (1984). Human Factors, Manpower, Personnel, and Training Clauses for the Concept Exploration and the Demonstration and Validation Requests for Proposal. (ARI RP 84-24, AD-A174-774). Ann Arbor, MI: Vector Research, Inc., Alexandria, VA: Army Research Institute for the Behavioral and Social Sciences,.

[20] Promisel, D. M., Miles, J. L., & Cherry, W. P. (1984). Human Factors, Manpower, Personnel, and Training Required Operational Capability (ROC) Enhancement (ARI-RP-84-23, AD-A169 945)). Alexandria, VA: U.S. Army Research Institute for the 8ehavioral and Social Sciences.

[21] Hardman Implementation Section (1986, March). HARDMAN Data Item Descriptions. Washington, D.C.: Chief of Naval Operations (OP-111).

[22] Campion, M.A. & Thayer, P.W. (1985). Development and Field Evaluation of an Interdisciplinary Measure of Job Design. J. Applied Psychology, 70 (1), 29-43.

[23] Edenfield, J.N. (1983, March). Aircraft Maintenance Specialty Classification: Aircraft Specialists versus System Specialists (Report No. 83-0635). Maxwell AF8 AL: AFSC/EDCC. A thesis submitted to the Air Command and Staff College, Air University (ATC, Maxwell AF8 AL.).

[24] Akman Associates, Inc. (1987, May). RAMPARTS Handbook: A Blueprint for Enhanced MPTS Planning. Silver Spring, MD: Akman Associates, Inc.

[25] Promisel, D.M., Hartel, C.R., Kaplan, J.D., Marcus, A., & Whittenburg, J.A. (1985). Reverse Engineering: Human Factors, Manpower, Personnel, and Training in the Weapon Systems Acquisition Process (ARI-TR-659). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences.

[26] Hartel, C.R. & Kaplan, J.D. (1984). Reverse Engineering of the BLACK HAWK (UH-60A) Helicopter: Human Factors, Manpower, Personnel, and Training in the Weapon System Acquisition Process (Research Note 84-100). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences.

[27] Arabian, J.M., Hartel, C.R., Kaplan, Jr. D., Marcus, A. and Promisel, D.M. (1984). Reverse Engineering of the Multiple Launch Rocket System: Human Factors, Manpower, Personnel, and Training in the Weapons System Acquisition Process (Research Note 84-102). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences.

[28] Daws, R.N., Keesee, R.L., Marcus, A., Hartel, C.R., & Arabian, J.M. (1984). Reverse Engineering of the STINGER Air Defense Missile System: Human Factors, Manpower, Personnel, and Training in the Weapons System Acquisition Process (Research Note 84-103). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences,.

[29] Marcus, A. & Kaplan, J.D. (1984). Reverse Engineering of the MI Fault Detection and Isolation Subsystem: Human Factors, Manpower, Personnel and Training in the Weapons System Acquisition Process (Research Note 84-101). Alexandria, VA: U.S. Army Research Institute for the 8ehavioral and Social Sciences.

[30] White, T. A. (1981). Manning of Recently Fielded Systems: Case Study of the Air Force E-3A (AWACS) (Report No. LMI-ML001, AD-8099 970L). Bethesda, MD: Logistics Management Institute.

[31] Interservice/Industry Simulator Cost and Lead Time Working Group (1987). Final Report, Simulator Cost and Lead Time Studies. Orlando FL: Naval Training Systems Center.

[32] Letter from HQ U.S. Army Materiel Command (General Richard H. Thompson) to Army Project Manager, Training Devices (PM TRADE) (31 March 1987). Design-for-Discard in Lieu of Repairs.

[33] Stephenson, R. W. and Stephenson, M. K. (1983). Design Requirements for Decision Support Systems for RDT&E. Information Processing and Management, 19 (6), 391-397.

[34] Thomas, E. L. Jr., and Deem, R. N. (1984, March). Unified Database Development Program. (AFHRL-TR-83-52, AD A140309). Wright Patterson AF8 OH: Logistics and Human FActors Division.

[35] Headquarters, U.S. Air Force (1985, Feb). Reliability and Maintainability of Air Force Weapon Systems-ACTION MEMORANDUM. Air Force, Chief of Staff and Secretary of the Air Force Memorandum for all Major Commands and Separate Operating Agencies.

[36] Analytic Services, Inc. (1985, October). Reliability and Maintainability Principles. Arlington, VA: Author.

[37] Sicilia, G.T. (1987, May). Training and Performance Data Center Capabilities. In Proceedings, International Conference on Military Personnel and Training, in press.

ABOUT THE AUTHORS

DR RO8ERT W. STEPHENSON received his Ph.D. in industrial psychology from the University of California in Berkeley in 1962. From 1957-67, he held positions as Assistant Director and Director of employee development and research groups at the Naval Weapons Center, China Lake, CA. From 1967-76, he was a Senior Research Scientist with the American Institutes for Research (AIR) at Silver Spring MD and Washington DC. While at AIR, he conducted and managed R&D on the design of military manpower, personnel, and training systems. He joined the Air Force Human Resources Laboratory in 1976, where his first assignment was as Chief of the Job Analysis and Job Requirements Branch. In 1979, Dr Stephenson joined the AFHRL Headquarters staff, where he has specialized in the analysis and evaluation of long range plans for manpower, personnel and training R&D.

LT COL FRANK C. GENTNER is currently Chief, Analysis Division, of the Manpower, Personnel, and Training Directorate (ASD/ALH), Deputate of Acquisition Logistics, Aeronautical Systems Division (AFSC), Wright-Patterson AFB, OH. He is responsible for developing the analytic staff and techniques to be used by the Directorate and coordinating research needs with R&D Laboratories to facilitate integration of MPT issues into the weapon system acquisition process. He holds a Bachelors in Psychology and Masters in Rehabilitation Counseling from the University of Florida. In addition, he has taken post-masters Industrial Psychology courses at St Mary's University. He has served as a personnel officer; an ATC Technical Training course chief; director of training evaluation; occupational analyst, staffer, and planner at the USAF Occupational Measurement Center; and initial Chief of Plans for the Training Development Service.

WHAT'S HAPPENING AT ASD REGARDING MPT

Lieutenant Colonel Frank C. Gentner, USAF
Manpower, Personnel, and Training Directorate (ASD/ALH)
Acquisition Logistics Deputate
Aeronautical Systems Division (ASD), Air Force Systems Command
Wright-Patterson AFB, OH 45433

ABSTRACT

More emphasis needs to be placed on manpower, personnel, and training (MPT) factors earlier in weapon system acquisition. To accomplish this, a new directorate was created at the Air Force Systems Command's largest product division, Aeronautical Systems Division (ASD). The MPT Directorate was chartered as a model organization to study and recommend ways in which the Air Force's most expensive asset, people, can more fully affect weapon system design, particularly in the early phases of the acquisition process when design adjustments are made most economically. This paper discusses the MPT Directorate's implementation plan and the progress made to date. Included are (I) a brief summary of why the Directorate was established, (II) MPT integration problems to be solved, (III) MPT process objectives, (IV) the Directorate's mission and functions, (V) its proposed concept of operation, and (VI) time-phased actions that must be taken to meet program objectives. The procedures and analytic tools which may be used by the MPT Directorate to consider MPT issues in the design process are highlighted in the concept of operations section. These procedures and analytic tools could be applied by other Air Force organizations in pursuit of similar objectives.

SECTION I

REASONS FOR ESTABLISHING THE MPT DIRECTORATE WITHIN ASD

While the United States Air Force (AF) led development of advanced analytic tools and data systems valuable for assisting MPT integration, it has not consistently applied available tools and data systems within a coherent framework. These tools, data systems, and techniques include the Advanced Personnel Data System (APDS), Comprehensive Occupational Data Analysis Programs (CODAP), Qualitative and Quantitative Personnel Requirements Information (QQPRI), and the Logistics Composite Model (LCOM), together with a host of logistical, safety and engineering data systems with MPT potential. The following section describes some of the missed opportunities which resulted in HQ USAF, HQ ATC, and HQ AFSC signing a memorandum of agreement in March 1986 to create the MPT Directorate within ASD as a model organization.

Growing concern in Congress, the Department of Defense and elsewhere in the weapon system acquisition communities has focused attention on MPT factors in the WSAP. Although many attempts have been made to emphasize the importance of MPT considerations, MPT integration into the WSAP has been a "sometimes thing". The insufficient attention being given to MPT factors while weapons systems were being developed resulted in MPT nightmares such as inefficient utilization of personnel, unnecessary expenditures for interim contractor support, late-to-need training development, and career ladder structures which affected Air Force readiness. Beginning in the early 1980s, a number of studies and reports served to bring MPT problems into clearer focus. Prominent sources included:

- Defense Science Board Study in 1982
- Air Force contract studies
- GAO reports
- Air Force Functional Management Inspections (Jun 1986)
- Congressional action
- Secretary of Defense action

Findings from these sources and the realization that the Services were manpower-constrained intensified the sense of urgency to improve integration of MPT factors into the Air Force WSAP. Meanwhile the Army and Navy established MPT-oriented programs, MANPRINT and HARDMAN, respectively. Significant high-level attention was directed at the timely acquisition and effective use of training devices for aircrew and maintenance personnel for new or modified weapon systems. This attention resulted in numerous conferences, reports, and studies to help improve training system acquisition, support and use.

The Defense Science Board Report, Dec 1982, stressed the need for more emphasis on effective training for operation and maintenance of future weapon systems. This report also addressed the problem of late delivery of training devices to operational units and the need to identify training requirements much earlier in the weapon system development process.

Starting in 1982, HQ AF/Director of Personnel contracted with Akman Associates, Inc., to study MPT in the WSAP. This study reported that the sizable increase in MPT support costs had amplified the need for earlier WSAP attention to identifying the manpower requirements and the associated MPT support costs of the new weapons system. Air Force planners needed to improve their capability to adequately anticipate the MPT needs associated with new

systems, react to those needs, and influence design decisions during the conceptual or design phase. In April 1983, the final report titled, "Enhancing Manpower, Personnel and Training Planning in the USAF Acquisition Process," was presented to the Deputy Chief of Staff, Manpower and Personnel. The study examined current MPT policies and practices, and recommended improvements to the WSAP. The report also included a plan to implement those recommendations.

In an ensuing contract, Akman Associates developed a basic concept for a manpower, personnel, and training information system. A draft report entitled, "System Concept Document for Manpower, Personnel, and Training Integration System" (MPTIS) documents Akman's efforts to date, illustrating the need to make MPT information available to key MPT players and acquisition personnel early in the WSAP.

Increased Air Force Inspector General's interest resulted in a Functional Management Inspection (FMI) during the period 13 May 1985 - 25 Jun 1986. The inspection examined the Air Force Simulator Certification (SIMCERT) Program. To assess the effective use of aircrew training devices, trainers supporting the F-15, C-5, F-16, F-111, EF-111A, C-141, and B-52 systems were evaluated. Training equipment for the ground launched cruise missile (GLCM) and maintenance training devices supporting these aircraft were also evaluated. This extensive FMI included maintenance training equipment covering visits to 11 ATC Field Training Detachments (FTDs) and the 3306th Training Evaluation Squadron (TES) at Edwards AFB (Flight Test Center), California.

The FMI concluded with the following findings and recommendations:

Management of training equipment issues requires continued emphasis. Early acquisition documents, such as, Statement of Need (SON), Program Management Directives (PMD), and Training Development Plans (TDP) should contain specific training needs information. Newly assigned training device acquisition and management personnel should attend a formal acquisition training course within the first six months of assignment.

Insufficient Manpower, Personnel, and Training (MPT) planning in the acquisition process degraded the quality and effectiveness of aircrew and maintenance training. The AF should improve MPT analysis concurrent with weapon system conceptual planning and include this requirement in initial PMD. A revision to AFR 50-11 should require early and accurate validation of aircrew and maintenance training device needs through a thorough MPT analysis.

Inadequate logistic support planning for training devices diminished availability for aircrew and maintenance personnel use. Another problem involved the lack of adequate coordination of engineering change proposals (ECPs) and training device managers, which often resulted in delays in delivery of trainers. Improved provisioning of training device spare parts based on their utilization frequency. Also, more effective coordination of modifications between Air Logistics Center system program managers and training device managers was necessary. In addition, AF should establish procedures to evaluate MPT issues resulting from weapon system modifications. Finally, proper interface with parent aircrew or maintenance training devices should be ensured by requiring software program validation for training systems before software is returned to training use.

The FMI concluded that early training planning in weapon system acquisition was essential. Further, to be successful, this planning must be developed in concert with weapon system manpower and personnel considerations. <u>M, P and T are totally interrelated and dependent upon each other</u>. MPT integration is needed to better ensure that quality training is provided to those who operate and maintain new and modified weapon systems. If this planning is inadequate, MPT needs associated with the weapon system will be ill-defined--increasing MPT costs, which are estimated to be some two-thirds the life-cycle costs of a weapon system.

The FMI also concluded that <u>Air Force MPT direction was highly decentralized with no organization responsible for integrating and monitoring system-related MPT requirements</u>. Further, early and more complete MPT analysis of system needs could have better assured procurement of more cost-effective and worthwhile training systems. This integration could be achieved, in the words of the FMI, through more effective program advocacy at Air Staff and MAJCOM level.

As a result of these and other reports, investigations, and studies, the March 1986 MPT Memorandum of Agreement (MOA) was signed and the process of staffing the MPT Directorate was begun. By September 1986, 12 personnel had arrived at Wright-Patterson AFB, OH and the MPT Directorate opened its doors. The MPT Colonel-level Steering Committee, consisting of representatives of the MOA's signatories, met to provide initial guidance. Since that time the assigned manpower, personnel, training and analyst personnel have engaged in learning the WSAP and the many acquisition tools and techniques already available at ASD. In addition, they developed, coordinated, and published the ASD/ALH implementation plan.

Meanwhile, interest in improving MPT integration has mounted. Congressional interest has been growing. This interest was typified by the action initiated by the Senate Armed Services Committee during deliberations on the FY 87 DoD Budget Request. The Committee expressed concern over an inability to properly consider the manpower requirements associated with major defense acquisition programs during the development and procurement decision stages of these programs. This lack of accurate information has resulted in Congress being confronted with unexpected requests for Service end strength increases to permit the sufficient operation, maintenance, and support of new weapons systems after the systems have been deployed.

The Committee suggested that in order to permit Congress to make more knowledgeable decisions concerning development and acquisition of such systems, the Secretary of Defense (SECDEF) should submit a report to Congress at least 90 days in advance of decisions regarding full-scale engineering development (Milestone II) and production of weapon systems (Milestone III). This report includes the complete manpower requirements associated with such systems, along with a plan for full operational deployment of such systems if no increase in personnel were authorized for the periods when deployment would occur. Language mandating this requirement was included in the National Defense Authorization Act for Fiscal Year 1987 (pp 165-166, HR 99-1001).

By Memorandum dated 14 Oct 1986, the Deputy Secretary of Defense (DEPSECDEF) advised the Secretaries of the Military Departments that DoD needed to improve the consideration given to Manpower, Personnel, Training, and Safety (MPTS) during all stages of the defense weapon system acquisition process. More emphasis was needed on MPTS factors in designing new systems, and better analysis of the impact new systems will have on total force MPTS requirements.

The Assistant Secretary of Defense, Force Management and Personnel [ASD(FM&P)], was assigned responsibility for coordinating efforts to improve MPTS planning for new defense weapon systems. FM&P was tasked to provide guidelines for submission on MPTS information to the Joint Requirements and Management Board (JRMB) at its milestone reviews. Also, FM&P will chair an OSD-Service staff-level working group charged with developing a general approach to MPTS planning that could be applied to all systems subject to JRMB review. The Undersecretary of Defense for Acquisition (OUSD/AQ) will be represented on this group in addition to the Services and the OASD(FM&P).

In a 31 Oct 1986 Memorandum entitled "Analysis of the MPTS Aspects of Proposed Defense Systems", the ASD (FM&P) expanded the working group's charter to consider what information the JRMB needs on the MPTS aspects of proposed systems at each stage of the planning process. The working group's overall objective will be to develop a general approach to MPTS planning and reporting that can be applied to all systems subject to JRMB review.

Mounting Congressional, DoD, and AF Secretariat interest coupled with a clear definition of MPT integration problems, should help provide the resources necessary to solve these problems.

SECTION II

MPT INTEGRATION PROBLEMS TO BE SOLVED

The MPT integration problems needing solution are listed below. This summary is extracted from the MPT MOA and ASD/ALH implementation plan, and has been endorsed by the Colonel-level MPT Steering Committee, which oversees the Directorate's operation.

o The Air Force WSAP needs a systematic approach to properly manage the integration of MPT factors.

o MPT planning has generally been fragmented and ill-timed. One result is that actions to integrate MPT factors have not occurred early enough in the WSAP to influence the design. Usually, the major effort to analyze MPT impacts on weapon systems has taken place in the Full Scale Development phase after most life cycle costs are fixed.

o The MPT planning effort often was not comprehensive enough to ensure that all pertinent factors were included or accurately applied. The Air Force did not clearly define weapon systems MPT goals for contractors designing and developing weapon systems. The initial estimates of manpower requirements needed to be identified as accurately as possible at the outset, with full consideration given to the specialties that will be involved in the new system.

o Data for MPT analysis was not available in a usable format to analyze weapon system design and suggest trade-offs. On-line data transfer networks were not readily available to expedite data transfer.

o Existing MPT requirements analysis tools were segmented and lacked the capability to do complete analyses to support complete MPT management decision process.

o Training and training equipment were not being adequately or consistently funded, developed, and procured throughout the acquisition cycle concurrently with their weapon system.

o MPT management was highly decentralized with no organization responsible for integrating and monitoring system related MPT requirements. Little effective direction and guidance was given to system program managers on MPT issues by Air Staff and the implementing commands (FMI finding).

o Lack of controls over the MPT process resulted in duplication of effort, higher cost for weapon systems, and ill-defined and late-to-need aircrew and maintenance training equipment.

SECTION III

MPT PROCESS OBJECTIVES

To help the Air Force solve these problems and disconnects, the MPT directorate was chartered as a model organization with a mission to address these MPT process objectives contained in the MPT MOA and MPT Directorate Plan:

o The primary objective of the MPT process is to improve analysis and integration of MPT issues in the acquisition cycle. MPT efforts must fully reflect the concerns and planning efforts of the using and supporting commands as well as the implementing command. To attain this objective the following sub-objectives must be met.

- Ensure that MPT factors and constraints are developed for use during early acquisition phases, where the potential to provide an optimally supported weapons system is greatest.

- Formulate a plan to bridge the gaps so MPT is factored into the acquisition process along with cost, schedule, performance, reliability, and maintainability; thus providing a more realistic, economical life-cycle cost for any given system in procurement or modification. MPT is a factor of weapon system supportability and needs to be highlighted as such.

- Ensure that training planning, requirements analysis, and training equipment development and production are planned, coordinated, and funded.

- Establish data sources, analytical tools, and procedures which support MPT trade-off analyses in the design of weapon systems.

- Make MPT analyses available in a format which emphasizes life-cycle cost-effective use of critical manpower, personnel and training resources.

SECTION IV

MISSION AND FUNCTIONS OF THE MPT DIRECTORATE

The MPT Directorate's mission is to provide centralized planning, direction and control for all MPT elements in the ASD weapon systems acquisition process. Major functions include:

o Directs the development and implementation of plans, policies, and analytical tools to quantify MPT impacts on and of developing weapon systems. In this regard, the Directorate places special emphasis on front-end analysis to bring MPT into the forefront during initial design processes of Preconcept and Concept Phases of the WSAP.

- Reviews regulations, military standards (MIL STDs), and acquisition documents to ensure MPT integration is encouraged.

- Develops a detailed MPT management network plan for the ASD WSAP to identify the time-phasing and method of implementing critical MPT elements.

- Assists in defining MPT roles and functions of key ASD organizations to ensure full MPT integration in ASD's everyday conduct of business.

o Employs analysis techniques and applicable policies and procedures to ensure consideration of alternative MPT utilization concepts and systems designs, and encourages necessary trade-offs to optimize cost and force effectiveness.

o Maintains contact with Systems Program Offices, Deputates for Development Planning, Training Systems, Engineering and similar organizations to assure the Directorate's participation in their studies and analyses, ensuring MPT integration is included early and throughout the systems concept and design stages.

o Maintains liaison with research organizations within and outside the military to optimize use of existing MPT resources; to promote research into areas beneficial to MPT; and thereby minimize costs for developing new models and systems to meet the organization's needs.

o Functions as the ASD focal point in directing the activities of MPT analysts who are assigned to SPOs to advise, assist and provide technical information, appropriate analytical models, methodologies and information systems to support the MPT planning process on the selected weapon systems.

o Provides the direction and leadership to obtain information systems equipment, software and related resources needed to support the Directorate's operations and MPT analyses.

o Advises the ASD Commander and HQ AFSC through the Deputy for Acquisition Logistics (ASD/AL) on MPT matters.

o Advises the MPT Steering Committee on the Directorate's progress in meeting the organization's objectives and obtains appropriate feedback for follow-on actions.

o Maintains liaison with key AF MPT organizations, such as AFMPC, AF Manpower Engineering Agency, ATC, USAFOMC, and other Steering Committee offices to exchange MPT information, data and concepts.

o Serves as the "model" Air Force MPT integration organization. Ensures that a comprehensive management plan is developed and updated with steps to achieve the organization's mission. The plan will provide detailed management information to similar organizations which may be established at the other AF product divisions. Further, ensures that MPT integration efforts are tracked by audit trails and evaluated to determine most effective methods and tools.

o Publicizes MPT integration issues and successes using available publication media including news releases, news letters, briefings, etc.

SECTION V

MPT DIRECTORATE PROPOSED CONCEPT OF OPERATION

INTRODUCTION TO CONCEPT OF OPERATIONS

Since its inception in September 1986, the MPT Directorate has been conducting studies and interacting with ASD organizations to establish roles and develop its concept of operation. In its 6-8 April 1987 meeting the MPT Steering Committee approved the MPT Directorate's draft plan, including the above mission and functions. In addition, the Committee endorsed briefings on ASD/ALH's preliminary concept of operation. Many aspects of this concept of operation will have to be fully staffed before they become codified.

The remainder of this paper is intended to present a glimpse of the proposed ASD/ALH concept of operation. We believe that open discussion of the proposed concept of operation will stimulate the feedback necessary to construct the most effective MPT organization. The concept of operation described below also includes the analysis tools and procedures which the MPT Directorate is strongly considering incorporating into ASD WSAP procedures.

To describe the MPT Directorate's concept of operations, it is necessary to examine many facets of ALH's given and planned activities. First, we'll look at the organizational placement and structures which govern and allow the Directorate to accomplish its mission. Second, since ALH must integrate its functions into the existing WSAP, we'll look at some of the critical existing and proposed documents which can allow MPT to influence design. Third, the Air Force already has a number of analytic tools and data bases which can be used in the WSAP. We'll examine a sample of these and the existing management tools which will give MPT visibility and help institutionalize MPT within the WSAP. Finally, since necessary actions differ during the various phases of the WSAP, we'll look at ALH's major proposed functions during the process.

FUNCTIONAL ALIGNMENT AND ORGANIZATIONAL STRUCTURE

Functional Alignment of ASD/ALH. The MPT Director is responsible to the Deputy for Acquisition Logistics (ASD/AL) for the proper functioning of the MPT Directorate (ALH) and oversight of activities of personnel matrixed to the program offices. Matrixed personnel are responsible to the Director for Manpower, Personnel, and Training, ASD/ALH, for MPT functions within ASD Acquisition Deputy organizations. This structure provides a central group of MPT personnel with visibility across program lines to ensure support and organizational objectives are attained; provides expertise for specific programs and/or issues during high demand periods (preparation and review of RFPs, SOWs, ITOs, etc.); and provides collocated expertise to work the program-specific issues as the weapons system progresses through the acquisition process.

General Concepts. The MPT Directorate will develop suitable data bases and analytical techniques to establish MPT constraints for the design process in much the same manner as performance and cost parameters are currently utilized. As a specific system transitions through the acquisition process, MPT Directorate personnel will aid the Systems Program Office (SPO) in establishing an MPT baseline, and will ensure all affected agencies are aware of the impacts upon that baseline as design, operational, or maintenance concept changes are proposed. The Directorate will address MPT supportability in both new and present acquisitions by providing both management guidance and technical support.

In a management role, the MPT Directorate will provide a means to formalize the MPT activities within SPOs. It will review existing policy and develop further guidance where appropriate. An essential management goal is to consolidate existing MPT direction and to provide a focal point for coordinating this direction.

In a technical support role, the MPT Directorate will be responsible for providing program managers with MPT expertise. MPT analysts will advise, assist, and provide technical information, appropriate analytical models, methodologies, and information systems to support the MPT planning process. These tools will be used to compare, project, and assess different design options and operational and maintenance scenarios for their relative MPT impacts and life-cycle costs. Maximum use will be made of currently available methodologies and tools such as Logistics Support Analyses (LSA), LSA Records (LSAR), appropriate LSA output reports, LCOM and Cost Oriented Resources Estimating (CORE) model outputs, and other life-cycle costs and MPT models. The MPT Directorate will develop contractual document statements which ensure that contractors are responsive to MPT concerns, and standards for evaluating MPT proposals during source selection.

Plan/Roadmap. The MPT Directorate educated its staff of manpower, personnel, training, and analyst personnel in the WSAP methods, procedures, tools, strengths, and weaknesses. As this study of MPT in the acquisition process progressed, the Directorate developed an implementation plan which describes its approach to MPT integration. In essence, this plan shows how the Directorate will develop its concept of operation. The plan received Air Force-wide feedback from offices concerned with MPT integration and was approved by the MPT Steering Committee on 8 April 1987.

Essentially, the model program will be conducted in two distinct, yet interdependent stages: (1) MPT Development, and (2) MPT Implementation. The MPT Development stage encompasses the effort required to research, design, plan, and test an MPT strategy within the existing WSAP; this is the stage in which the Directorate is now operating. The MPT Implementation stage takes the MPT strategy and implements it in the major weapon system acquisition process.

MPT Steering Committee. In consonance with the MPT objectives established in the MPT MOA, the Colonel-level Steering Committee provides a working forum for development of the model MPT program. It provides guidance, feedback, planning assistance, visibility, helps work issues, and assesses progress. It conducts semi-annual reviews of all aspects of the model program and provides assessments to the MOA signatories.

ALH SPO-Matrixed Personnel. ASD/ALH will matrix an MPT analyst to selected major weapon system SPOs. The matrixed MPT analyst will work for the Director of Logistics (DOL) to ensure MPT issues are fully addressed. Assisting the DOLs with manpower, personnel, and training expertise in working the Integrated Logistics Support (ILS) plan, they will support the Chief Systems Engineer in identifying the full MPT ramifications of the various design issues faced by the SPO. They will work interactively on all design issues having MPT implications, closely with SPO-matrixed engineering and logistics personnel. They will be trained in manpower, personnel, training, WSAP, and basic MPT analysis techniques. When ALH is fully operational, approximately half of its 36 personnel will be matrixed to SPOs.

MPT Directorate Home Office Support. ASD/ALH's home office will select, train and matrix the MPT analysts to selected SPOs. ALH will also provide continuing training, advice and counsel. The home office will be responsible for career enhancement, and supporting them on the more complex MPT analyses. For those SPOs not having matrixed personnel, ALH will provide staff assistance and MPT analysis on a space available basis.

Coordinating and Assisting ASD Organizations. Several key ASD organizations will have a special relationship with ASD/ALH. These include ASD's Deputy for Engineering, Safety Office, Deputy for Training Systems, the Deputy for Development Planning, and the ATC Operating Location (OL) at ASD.

Deputy for Engineering (EN). Like ALH, EN also has matrixed engineers in the SPOs who are supported by the home office. Close coordination between ASD/ALH and EN needs to be maintained to ensure that MIL STDs and procedures are developed which encourage MPT influence on design. Interaction will be especially close with EN human factors, training, and LCOM personnel.

Safety Office (SE). SE also has matrixed system safety engineers and managers in the SPOs who are supported by their home office. Close coordination between ASD/ALH and SE will be maintained to ensure that safety programs and analyses are developed which encourage MPT influence on design. Interaction will be especially close with safety training and design hazard analyses to identify high drivers.

Deputy for Training Systems (YW) and the ATC Operating Location (ASD/TTGT). ALH will also coordinate closely with YW (formerly SIMSPO) and with

ASD/TTGT (ATC OL) to ensure training systems developed to support aircraft systems LSA and engineering data are available in sufficient time to develop training systems that will be fielded with the weapon system.

Deputy for Development Planning (XR). ALH will be active during the WSAP phases in which design can be most influenced. Therefore, ALH will participate with the Development Planning Deputate on Long-Term Planning Projects that have potential MPT implications. Examples of these projects include the High Reliability Fighter, Embedded Trainer Concept, Recce-Attack Fighter Training System, etc. Matrixing an MPT analyst to this Deputate can help ensure that MPT factors are considered early in the acquisition process.

Collocate MAJCOM OLs. The MPT Steering Committee accepted ALH's recommendation that using MAJCOMs establish operating locations (OLs) at Wright-Patterson AFB for system-manpower analysis for staffing. These OLs, if implemented, will work as a team with ASD/EN, ALH and their SPO-matrixed personnel to conduct LCOM and other MPT analyses, and provide using MAJCOM inputs, guidance and coordination to determine the manpower assessment. Under this concept the SPO would be responsible for consolidation of all manpower assessments, including AFLC and ATC inputs, and will provide them to the using MAJCOM. The using MAJCOM will be responsible for forwarding a consolidated statement of manpower requirements to HQ USAF prior to Milestones II and III to satisfy the 1987 Defense Authorization Bill's reporting requirements. The joint ASD/EN, ALH, and using MAJCOM team will have the best chance of achieving an optimum balance between manpower and equipment costs and of influencing design. Also, developing the figures jointly offers the best opportunity for deriving an unbiased estimate. Strategic Air Command (SAC) has already established an OL at Wright-Patterson AFB, OH for this purpose. HQ USAF is staffing this concept with other MAJCOMs.

WEAPON SYSTEM ACQUISITION DOCUMENTS

The Weapon System Acquisition Process (WSAP) is a complex set of procedures controlled by several key documents. ASD/ALH plans to integrate MPT by reviewing, recommending changes, and ensuring the MPT aspects of these documents are prepared in sufficient time to affect design. These reviews will be conducted in conjunction with ASD/EN/YW/SE/XR counterparts.

Early Acquisition Requirement Documents. ALH will review early acquisition documents to ensure MPT requirements and issues are appropriately addressed. The goal is to be proactive in setting specifications for design which require MPT trade-off analyses to construct systems which comply with MPT constraints. ALH will work with Air Staff, using MAJCOMs, the ASD staff, ATC, AFMPC, and ASD/EN (as appropriate) to ensure MPT constraints and goals are clearly communicated in early acquisition documents.

These documents include the Statement of Operational Need (SON), System Operational Requirements Documents (SORD), Requests for Proposal (RFPs), and Statements of Work (SOWs). MPT constraints, goals, issues, analyses and concerns need to be included in these documents if design is to be affected. Ultimately, the prime contractors must see the benefit of applying good MPT techniques in design. For this reason, ASD/ALH plans to furnish representatives for source selection, and ultimately develop generic MPT criteria which can be used as a guide for source selection of individual systems.

Existing Acquisition Plans. In addition to acquisition documents and source selection, these acquisition plans include MPT information: the Human Factors Engineering (HFE) plan, the Integrated Logistics Support Plan (ILSP), the Training Development Plan (TDP) and the Test Plan. ALH intends to improve the accuracy and completeness of MPT information in these documents. Also, ALH will attempt to ensure that MPT information is entered sufficiently early to allow design changes.

Proposed System MPT Plan. One way to ensure integration of MPT goals and constraints is to develop a weapon system MPT plan prior to Milestone 0. This kind of document could be developed by a team of MPT players to focus attention on MPT issues. The Systems MPT Plan could be a living document produced/revised (as a minimum) before each milestone, examining MPT issues which come to light as more is learned about the system. It would be designed to coordinate M, P, and T issues and goals for the system into an integrated approach. Further, this proposed plan would address ways in which MPT will be integrated into system design considerations.

Establishes MPT Constraints & Provides Visibility. In the initial phases of acquisition, the System MPT Plan could document the MPT constraints and goals for the system. Later, the plan could document trade-off decisions between M, P, and T, and between MPT and system performance/cost/schedule and other considerations, giving visibility to MPT issues. The MPT plan could be coordinated among major MPT and system players, and be coordinated with the Human Factors, System Safety, Integrated Logistics Support (ILS) and Training Development Plans (TDP).

Add M & P to TDP. Because there are so many acquisition plans, often presenting the same or similar information, it would be best to avoid proliferation of plans. It is possible to expand an already existing plan, the TDP, for MPT purposes. The System MPT Plan could be composed of M and P issues added to the Training Development Plan. The Systems MPT Plan would start before Milestone 0, much earlier than the Training Development Plan (TDP) currently is begun. Expanding the TDP appears to make sense, since many critical MPT issues are already required in the TDP, yet are not developed soon enough to affect design.

Character Changes with Phases of WSAP. The character of the Systems MPT Plan would change as the weapon system matures. Initial editions would focus on MPT goals and constraints, with later versions focusing on system trades and a full-fledged Training Development Plan.

System MPT Planning Team. A System MPT Planning Team would be established for selected programs before Milestone 0. The team could be chaired by ALH until the SPO is staffed. Since the SPO Director is chair of the Training Planning Team (AFR 50-8), this Director should probably chair the System MPT Team. He/she will likely use the ALH SPO-matrixed person to work MPT plan issues and to document the System MPT Planning Team recommendations.

Team Composition Changes with WSAP. Before Milestone 0, proposed members of the System MPT Planning Team include the ASD MPT Directorate, Developmental Plans Deputate, ATC, AFMPC, the Using MAJCOM, and Air Staff Manpower and Personnel Plans. After staffing of the SPO (around Milestone 0), ASD Engineering and MPT Directorate SPO-matrixed personnel, the Deputy Program Manager for Logistics (DPML), and eventually the Prime Contractor(s) join, as ASD Developmental Plans and Air Staff agencies reduce their involvement. Later in the acquisition process, the team may also include the Air Force Operational Test and Evaluation Center (AFOTEC) to ensure appropriate test planning of MPT issues. The initial Systems MPT Plan can furnish needed input for the TDP and the MPT parts of the ILS plan. As the Training Planning Team is formed, meetings could be held in conjunction with the System MPT Planning Team meetings.

Regulatory Guidance. Finally, ALH plans to review the host of acquisition, manpower, personnel and training regulations to determine consistency and their impact on MPT integration. In addition, the Military Standards (MIL STDs) or Military Primes (MIL PRIMES) and Data Item Descriptors (DIDs) furnish guidance to contractors and writers of contracts. A thorough review and necessary revision will greatly facilitate MPT integration.

Automated Design Tools as MIL STDs. In place of MIL STDs, the new Computer Aided Design (CAD) system human factor design tools like Crew Chief and COMBI/MAN can impact on prime contractor engineers while they are in the design process. These human factor automated design aids use a three dimensional manikin on the CAD screen, allowing engineers to "see" accessibility problems. By making these systems available to all prime contractors and requiring their use, many of the access problems experienced in the past can be overcome during the design process. ALH will encourage their use and standardization into the design process.

USE EXISTING MPT ANALYSIS TOOLS AND DATA SOURCES/SYSTEMS

The Air Force has many valuable MPT analysis tools and data bases/systems which, if integrated and applied, could pay big dividends. For example, LCOM (the AF-approved manpower modeling) has been thoroughly validated to produce aircraft maintenance manpower assessments through simulation. For aircraft maintenance, this method is far superior to and more accurate than the Army and Navy HARDMAN procedures of adding task times. LCOM uses the Maintenance Data Collection (MDC) system to furnish the crew size, frequency and maintenance tasks times for each aircraft system. Before inputting these data into LCOM, the MDC data are operationally audited by manpower personnel for consistency and accuracy. MDC data offers hard maintenance data on predecessor systems, which can be used in LCOM simulations of future systems. The Advanced Personnel Data System (APDS) and Occupational Survey Program furnish comprehensive data on which tasks are performed by Air Force personnel. These data are available "on line" through the Occupational Research Data Bank. Logistics Support Analysis (LSA) and LSA Record (LSAR) are available from prime contractors, and could be requested sufficiently early to provide training developers with needed information in time to field training systems with the aircraft. Validated human factor tools, which have been applied for years to cockpit design, could be applied more fully to maintenance. And both safety and human factor data bases could be used to focus attention on MPT high drivers in time to affect design. Let's examine a few of these tools/data bases to obtain an image of how ALH envisages employing them.

LCOM as an Iterative Process and Source of Maintenance Manpower Estimates. LCOM is used as a tool to identify MPT high drivers and aircraft maintenance manpower assessments throughout the WSAP. LCOM will be conducted at increasing levels of specificity from Preconcept sensitivity analyses on the predecessor system, to baseline system comparisons during Concept Development, to LCOM models based on HFE/LSA comparability input data during the Demonstration/Validation Phase and more final HFE/LSA/engineering data during Full Scale Engineering Development. It will be the recommended source of aircraft maintenance manpower estimates. ALH will be responsible for conducting or contracting the LCOM sensitivity analyses using predecessor system or generic data prior to the SPO being formed. Once the SPO has been formed, EN may conduct/contract for LCOM analyses on selected major weapon system as requested by the SPO. The LCOM model produced by the ASD/EN, ALH, SPO and MAJCOM OL personnel could be used for both manpower assessments and engineering analyses.

APDS and CODAP Available through the ORDB. The Air Force has one of the most sophisticated personnel data systems available in any Service. It can describe AF military personnel in great detail by career ladder and has great flexibility for data manipulation to answer MPT questions. Comprehensive Occupational Data Analysis Programs (CODAP) enables the USAF Occupational Measurement Center to describe the tasks and related task data about each AF career ladder. Again, these data are stored in a very flexible format. Both APDS and CODAP data bases feed the Occupational Research Data Bank (ORDB) which makes computer "runs" available through "on-line" modem access. The one drawback to these data is that they are presented by Air Force Specialty Code (AFSC), which may or may not be directly related to an aircraft system. Research is under way to alleviate this disconnect. Once this is accomplished, the rich AF personnel data resources can be made available for the personnel/AFSC analysis needed to input to the LCOM manpower analysis. Until that time, ALH hopes to use this data as best possible in a manual mode.

LSA/LSAR and HFE Task Data Ordered Iteratively, and Source for Training Development. LSA/LSAR and HFE task data will be ordered iteratively throughout the WSAP beginning with milestone 0 to provide data for addressing MPT issues. It is critical that the users tailor what they need, specify when each delivery is to take place, and that the medium specified is the most efficient/effective for immediate use. Later in the WSAP, complete and accurate LSA/LSAR is needed to address training development issues. If it is late, the training development system cannot ensure adequately trained personnel for the Initial Operational Capability (IOC). In the past, LSA/LSAR has received low priority. ALH will attempt to give higher visibility to HFE and LSA through the System MPT Plan so that the Air Force buys a total weapon system, including everything necessary for trained personnel to operate, maintain, and support it when fielded.

Human Factoring Maintenance Tasks. In the past, most human factors efforts centered around the cockpit

and crew positions. Because of the expense and lessened availability of maintenance manpower, it is critical that we start assigning the same human factor priority to maintenance as we have crew positions. The result will be maintenance tasks and equipment which reduce man-hour requirements per unit, and increase the efficiency of maintenance turn-around and wartime readiness of new systems.

Coordination with Safety and Human Factors Initiatives. Because some safety and human factors issues can have large MPT implications, close coordination with the ASD Human Factors Engineering (HFE) Branch and Safety Office is appropriate. Both human factors and safety data sources could help identify high driver MPT issues. ALH will work closely with these offices to develop data-based ways of identifying MPT issues of consequence. By supporting issues of common interest, the three offices have a better chance of obtaining SPO funding for necessary analyses and of gaining implementation of HFE, Safety, and MPT positions.

MPT MANAGEMENT ACTION

In addition to the above methods, which may have management implications, ALH plans to conduct the intensive search and development to develop a CSNAS (see below) Network for MPT. In addition when fully staffed, ALH will take on the mission of developing a three-tiered Systems MPT course, and will work with Air Force Institute of Technology (AFIT) and ASD acquisition courses for full inclusion of MPT principles.

Development of CSNAS Network. ALH will conduct a thorough review of regulations, MIL STDs, MIL PRIMES, operating instructions, and MPT requirements working cooperatively with appropriate ASD, AFSC, AFALC and Air Staff organizations. Based on the findings of this review ALH will construct a Computer Supported Network Analysis System (CSNAS) model network. This computerized "PERT diagram" provides SPOs with a government owned project management tool which meets regulatory requirements for each major program to have a networking system. The computer network identifies critical actions over the development of each system, and allows the SPO to show impacts of program changes on the schedule using the critical path method. The MPT model that ALH will develop will provide SPO personnel with a prototype of what they should accomplish regarding MPT in their program. They can tailor the prototype model to meet the unique facets of their program so that MPT becomes an integral part of their program's management. Thus the MPT CSNAS model will provide guidance on the timing and procedures for implementing MPT within each program.

Systems MPT Course. While attempts have been made, it is clear that a course which fully covers manpower, personnel and maintenance training (as well as operator training) needs to be developed. As funds and manpower become available, ALH plans to monitor the contract for a three-tiered MPT course. A short version would be tailored for senior AF and industry. A more detailed and functionally-oriented course would address the needs of MPT managers, logistics, human factors, and other Air Force and industry personnel. Finally, a detailed MPT analyst course is needed for the ALH, SPO-matrixed, and other personnel who are expected to conduct weapon systems MPT analysis.

Update AFIT and ASD Acquisition Courses. Working with AFIT and ASD training personnel, ALH hopes to update the courses used to educate and train ASD personnel on the acquisition process. As regulatory guidance is published, these courses are normally updated. Therefore, as progress is made in updating regulations and MIL STDs, they will be incorporated into the AFIT courses. In addition, the Directorate will furnish assistance in updating MPT integration aspects of AFIT WSAP courses.

WSAP TIMELINE ACTIONS

Another way to describe ALH's concept of operations is to look at the major functions ALH will perform, or ensure are performed by acquisition phase. These are the MPT Directorate's goals:

Preconcept: Develop MPT constraints and goals, define problems to be solved with new system, identify MPT analyses and trades to be examined, examine new technologies, participate in planning projects, and develop source selection criteria.

Concept: Explore MPT alternatives, examine implications of design trades, develop alternate MPT concepts, influence design, and develop defined source selection criteria.

Demonstration/Validation: Evaluate MPT implications of alternative systems, recommend changes, refine MPT concept, refine manpower assessments, order data for training development, plan MPT tests, and develop source selection criteria.

Full Scale Development: Evaluate System for MPT issues, finalize and publicize MPT concept, ensure training system developed, finalize manpower estimate, and conduct/evaluate MPT tests.

Production/Deployment: Review test results for MPT implications, develop MPT lessons learned, and validate MPT concept.

SECTION VI

DIRECTORATE PROGRAM OBJECTIVE TIME-PHASED ACTIONS

The following time-phased actions being undertaken by the MPT Directorate illustrate some of the activities initiated by ASD/ALH. They are being undertaken in support of the ALH mission and to implement the concept of operations, explained in Section V.

-Determine MPT roles and responsibilities

- Establish Safety Program interface

- Include MPT factors in regulations, LSA, DIDs, MIL STD/MIL Primes, procurement documentation formats, etc.

- Develop standard paragraphs and checklists for procurement documents

- Develop prototype Systems MPT Plan

- Develop procedures for MPT integration into source selection

- Develop MPT CSNAS network

- Develop MPT education and training course(s)

- Update MPT information in AFIT and ASD acquisition courses

- Develop MPT analysis capabilities using existing analysis tools and data bases

- Promote research and development to correct MPT analysis tool/data base deficiencies

- Establish information systems support for MPT Directorate

- Develop procedure to monitor force structure performance indicators (e.g., aptitude, education, retention, skill projections) for potential impact on developing weapon systems

- Develop/manage MPT program decision package

- Assess and procure contractual MPT support

- Develop matrix personnel plan

- Develop MPT information and publicity plan

SUMMARY

The Manpower Personnel and Training Directorate was established to test whether a long-standing need to fully integrate MPT considerations into the acquisition process could be institutionalized by a model organization at the AFSC product division level. The initial cadre of 12 military personnel developed an MPT Implementation Plan which is summarized above. The goal of the organization's concept of operation is to determine how existing MPT analysis tools, data sources and procedures can more fully improve consideration of MPT factors in the WSAP. The MPT Directorate is giving favorable consideration to using or ensuring use of LCOM, CODAP, HFE task analyses and CAD tools, and LSA/LSAR. In addition, it plans to review acquisition documents for inclusion of MPT requirements, and use CSNAS as a management prototype to encourage timely requesting of MPT analyses and data. Regulation and MIL STD changes coupled with a three-tiered Systems MPT course will help encourage acquisition mangers to consider MPT. And finally, a Systems MPT Plan is proposed to integrate M, P, and T and give MPT the appropriate planning and visibility. With this tool, the SPO-matrixed MPT analysts will be able to obtain MPT goals and constraints to influence system design, and can highlight the important MPT issues within and outside the SPO.

REFERENCES

Aeronautical Systems Division/Manpower, Personnel, and Training Directorate (1987, Feb). _ASD/ALH Implementation Plan (Draft)_. Wright-Patterson AFB, OH: Author.

Akman Associates, Inc. (1983, April). _Enhancing Manpower, Personnel, and Training Planning in the USAF Acquisition Process, Final Report_. Silver Spring, MD: Author (AD- F630514).

Akman Associates, Inc. (1986, Sep). _System Concept Document for Manpower, Personnel, and Training Integration System (MPTIS)_. Washington, D.C.: Author.

Cox, Chapman B. Assistant Secretary of Defense (Force Management & Personnel) (31 Oct 1986). Memorandum for Secretaries of Military Departments, Subject: _Analysis of the MPTS Aspects of Proposed Defense Systems_. Washington, D.C.: Author.

Defense Science Board (1982, Dec). _Report of the Summer Study Panel on Training and Training Technology_. Washington, D.C.: Office of the Undersecretary for Research and Engineering.

General Accounting Office (1981, January). _Effectiveness of U.S. Forces Can Be Increased through Improved Systems Design_ (Rep PSAD-81-17). Washington, D.C.: Author (AD A114237).

General Accounting Office (1985, September). _The Army Can Better Integrate Manpower, Personnel, and Training into the Weapon Systems Acquisition Process_. GAO/NSIAD-85-154. Washington, D.C.: Author.

United States House of Representatives, 99th Congress (1986, Oct, pp 165-166.) _National Defense Authorization Act for FY 1987, Conference Report_. Washington, D.C.: Government Printing Office (HR 99-1001).

Taft, William H., IV, Deputy Secretary of Defense (14 Oct 1986). Memorandum for Military Departments and Secretaries for ASD (Acquisition) and (Force Management and Personnel), Subject: _MPTS Planning for New Defense Systems_. Washington, D.C.: Author.

ABOUT THE AUTHOR

Lt Col Gentner is currently Chief, Analysis Division, of the Manpower, Personnel, and Training Directorate (ASD/ALH), Deputate of Acquisition Logistics, Aeronautical Systems Division (AFSC), Wright-Patterson AFB, OH. He is responsible for developing the analytic staff and techniques to be used by the Directorate and coordinating research needs with the Laboratories to facilitate integration of MPT issues into the weapon system acquisition process. He holds a Bachelors in Psychology and Masters in Rehabilitation Counseling from the University of Florida. In addition, he has taken post-masters Industrial Psychology courses at St Mary's University. He has served as a personnel officer; an Air Training Command Technical Training course chief; Director of Training Evaluation for the Defense Equal Opportunity Management Institute; occupational analyst, staffer, and planner at the USAF Occupational Measurement Center; and initial Chief of Plans for Training Development Service.

FOOTPRINT: ONE SMALL STEP FOR MPT

Major Joseph Fil
Armored Family of Vehicles Task Force
Ft Eustis, VA 23604

Kris Hoffman and Don Johnson
Defense Training and Performance Data Center
Orlando, FL 32826

Dennis Collins
Soldier Support Center - National Capital Region
Alexandria, VA 22332

ABSTRACT

Each of the military services are implementing new forecasting methods for Manpower, Personnel and Training (MPT) requirements of major new systems. The primary aim is to make better trade-offs in weapon system design and control MPT resource increases. Footprint is a project under the direction of the Soldier Support Center, in support of Manpower and Personnel Integration (MANPRINT) objectives. An integrated data base has been developed which will enable combat developers to quickly assemble pertinent MPT information early on in the weapon system acquisition process.

INTRODUCTION

In the summer of 1986, the Defense Training and Performance Data Center (TPDC) began the development of an automated MPT data integration technique under the sponsorship of the Army Soldier Support Center (SSC). The goals of this project, referred to as "Footprint," were to develop an automated tool in support of up-front analysis which utilizes existing data bases, and which could quickly display the training related characteristics of an existing weapon system or end item. The compiled data could be produced as a series of standard MPT reports aggregated either by predecessor system or by Military Occupational Specialty (MOS).

The underlying premise of Footprint is that the MPT profile of the predecessor system is the best data available prior to and during the early Concept Exploration phase. The importance of having a reliable MPT baseline during the Concept Exploration phase is underscored by studies which suggest up to 70% of the life cycle cost of a new weapon system is fixed at the end of this phase. Ironically, the greatest opportunity to influence the development of a new system occurs when MPT analysis has traditionally been at its lowest level of intensity.

While comparability analysis methods, such as HARDMAN, are useful techniques to forecast specific MPT impacts, they have a major shortcoming. Because HARDMAN is a complex and time-consuming process, it typically does not yield results until after various design options have been evaluated in detail. Although comparability analysis permits the DoD program manager to predict MPT impacts accurately, they may predict increased MPT requirements after it is too late to change system design without dramatic cost increases.

The Footprint methodology provides a comprehensive profile of predecessor systems at the earliest phase of the Weapon System Acquisition Process. Footprint MPT reports contain both historical and projected authorizations of selected MOS, along with many other training, performance and force structure issues. These reports provide a practical starting point for the development of the new system, since these are primarily the resources that will be impacted by its deployment. Although the MPT requirements will change as a result of a host of variables such as accessions, MOS restructuring and demands of the new program, there is a need to carefully manage these fluctuations in order to remain within resource limits.

In addition to individual service concerns about the MPT price tag of a new system, Congress has become increasingly active in 1986 and 1987. Section 1208 of the Defense Authorization Act of FY 87 makes it imperative that the services provide the Secretary of Defense and Congress with a comprehensive manpower estimate including training resources, prior to approval for entry into Full Scale Development of a major new system. This requirement coupled with on-going programs to speed up the acquisition process will place Program Managers (PM) in the difficult role of estimating, programming and controlling MPT resources to a much greater extent than ever before. The use of automated

NEW SYSTEM	# PRED	# MOS
Armored Family of Vehicles (AFV)	36	83
Advanced Anti-Tank Weapon System - Medium (AAWS-M)	1	3
Advanced Anti-Tank Weapon System - Heavy (AAWS-H)	4	5
Multi-Channel Communication Objective System (MCOS)	2	4
Frequency Hopping Multiplexer (FHMUX)	2	4
Electro-Optic Test Facility (EOTF)	9	6
TOTALS	54	105

Table 1. Relation of New System to Predecessors and Associated MOS

data base tools such as Footprint combined with existing MPT analysis techniques, will be essential in supporting new program planning and justification in the coming era of fiscal austerity.

SCOPE OF THE FOOTPRINT PROTOTYPE

The scope of the Footprint prototype was limited to integrating existing Army data sources to produce a series of standardized reports for 3 to 5 new system acquisitions. The Army Vice Chief of Staff, the Commander of the Army Soldier Support Center, and the Commandants of the Army Infantry and Army Signal Schools, were briefed on the Footprint concept. To test the concept they selected six new weapon systems, in the concept exploration phase, as candidates for the Footprint prototype. These new systems are shown relative to their number of predecessors and associated MOS in Table 1.

OVERVIEW OF THE ARMORED FAMILY OF VEHICLES

The AFV Task Force (AFVTF) located at Ft. Eustis Va, is headed by MG Robert J. Sunell, former Program Manager for the M1 Tank program. The objectives of the AFV program include developing and fielding a force capable of defeating the threat of the 1990's and beyond. The reduction of overall system and force Operation and Support (O&S) Costs is a primary goal. The AFV will be operated throughout the theater by combat, combat support, and combat service support units. The AFV fleet will be the basis of the total Army armored vehicle inventory from the mid 1990's to the next generation of AFV. The AFV will replace the entire range of currently fielded and projected armored vehicles through active Army, Reserve Component (RC), and Army National Guard (ARNG). The AFV will incorporate modularity, component commonality, common battlefield signature, common vehicle electronics architecture, and multiple system capabilities. There are currently 29 AFV roles and mission requirements within the emerging family concept. These are listed in Table 2.

Future Armored Combat System
Engineer SAPPER Vehicle
Directed Energy Weapon Vehicle
Armored Ambulance
Line-of-Sight Anti-Tank Vehicle
Advanced Field Artillery System Cannon
Multiple Launch Rocket System
Armored Maintenance Vehicle
NBC Reconnaissance System
Mortar Weapon System
Armored Recovery Vehicle
Future Command and Control Vehicle
Armored Security Vehicle
Combat Mobility Vehicle
Light FACS
Infantry Fighting Vehicle
Armored Reconnaissance Vehicle
Fire Support Team Vehicle
Line-of-Sight Air Defense Vehicle
Nonline-of-Sight Air Defense/Anti-Tank Vehicle
Combat Earth Mover
Armored Rearm Resupply and Refuel Vehicle
Armored Smoke Generating System
Combat Gap Crosser (Bridge)
Armored Medical Aid Station
Elevated Target Acquisition System
Intelligence and EW Vehicle
Combat Excavator

Table 2.
AFV Roles and Mission Requirements

In the words of an AFVTF spokesman, "The successful culmination of the AFV program will depend on the ability to create a common integrated perspective of one Army". The AFVTF is chartered with restraining the creation of additional MOS, and must strive to merge existing MOS. In fact, the AFV Justification for Major System New Start (JMSNS) specifies that, "No increase in manpower resources will result from the AFV program." This is particularly significant when considering that the AFV will directly impact at least 83 MOS associated with 17 proponent schools or up to 57% of the active duty Army population. Assessing the potential impacts on these personnel, and developing strategies to minimize the costs associated with the AFV MPT are issues directly supportable by Footprint.

DEVELOPMENT OF THE PROTOTYPE

Identification of Data Elements, Formats and Data Sources

With Army concurrence and support, TPDC proceeded with the identification of data elements, data sources, and report formats which would best help the AFV Task Force and Army PM's determine the MPT characteristics of the predecessor systems. The first step was to determine what information was required by MPT data users in the early phases of acquisition cycle. A series of interviews were conducted with a variety of Army personnel including Combat Developers, Training Developers, and Logisticians at Ft. Benning, Ft. Gordon, Ft. Bliss, Ft. Eustis, Ft. Knox, Ft. Belvoir and Ft. Lee. Table 3. summarizes the MPT data areas that were described as a high priority by the interviewees.

MANPOWER ELEMENTS
- Military Occupational Specialty (MOS)
- Additional Skill Identifier (ASI)
- Language Identification Code
- Manpower Authorizations
- Manpower Requirements

PERSONNEL ELEMENTS
- Primary MOS
- Duty MOS
- Gender
- Education Level
- ASVAB Composite
- AFQT Score
- Mental Category
- Standing Height
- Sitting Height
- Kneeling Height
- Functional Arm Reach
- Color Vision
- Acuity Vision
- Weight
- PULHES

TRAINING ELEMENTS
- Training Location
- Course Prerequisites
- MOS Required
- Class Size
- Annual Class Capacity
- Class Length
- Number of Graduates
- Instructor Contact Hours
- Tasks Taught
- Student/Instructor Ratio
- Training Type
- Number of POI Hours
- Number of System Specific Hours
- Number of Instructors

Table 3.
MPT Data Survey Findings:
High Priority Items

The survey also examined how the data was used in the system acquisition process in order to identify the most efficient means of displaying the standardized report formats. The results consistently identified the set of MPT data needed for completing the MANPRINT Target Audience Description (TAD). Other MANPRINT related events identified through the interview process which consistently require MPT data as inputs are Early Comparability Analysis (ECA), Human Factors Engineering Analysis (HFEA), HARDMAN Comparability Analysis (HCA), and New Equipment Training Plans (NETP).

After determining which MPT data elements and report formats were needed, it quickly became apparent that a large number of data sources would be necessary to provide complete support in the early acquisition phase. At this point, in order to remain responsive to the AFV milestones, the focus of the prototype effort shifted to fulfilling the greatest percentage of MPT data needs with the most appropriate subset of existing data sources. Data sources were reviewed for content, completeness, and accessibility. Of those Army and DoD sources identified, five were selected to demonstrate the capabilities of the Footprint prototype.

The selected data sources were the Army Training Requirements and Resource System (ATRRS), the Personnel Management Authorization Document (PMAD), the Army Enlisted Master File (EMF), the Military Entrance Processing Command (MEPCOM) Accession File, and Army Regulation (AR) 611-201. The EMF and MEPCOM tape extracts were obtained from the Defense Manpower Data Center (DMDC). Using the results of the Army interviews as a guide, key data elements were identified in each source.

Data Extracts, Programming, and Report Generation

After the data elements within each data source were identified and data file extracts obtained, the process of determining exactly how the required report formats were to be generated was addressed. Figure 1 is a conceptual view of how the separate file extracts were merged to form a single Footprint reference file. In the process of developing this reference file, several key file design issues were considered:

(1) Which data elements allow the merging of data from one file extract with data from another extract?

(2) How can the various file and data formats be used to form a single set of report formats while preserving the data integrity of each data source?

(3) What required MPT data elements are not visible in the file extracts, but can be discerned utilizing the existing data?

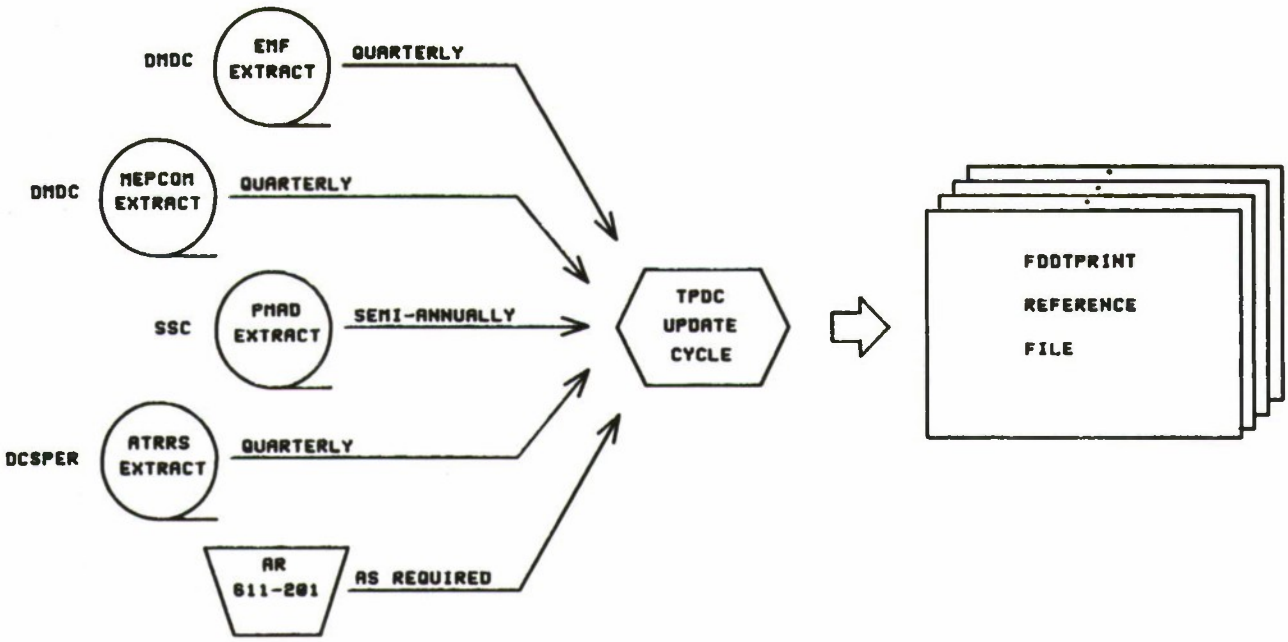


Figure 1. Footprint Reference File Update Cycle

(4) Based on the magnitude of the data file extracts (approximately one gigabyte or one billion bytes), how can the report formats be generated in an efficient and cost effective manner?

All of these issues were successfully addressed during the formulation of the Footprint reference file, and resulted in an integrated system that supports the generation of 27 unique report formats for any enlisted MOS.

Figure 1. also represents the "steady state" process of routinely receiving tape extracts (quarterly/ semi-annually) and updating the Footprint reference file. These tapes represent the most current automated data available, since the update cycle mirrors that of the data sources. In concept, any or all MOS reports could be produced systematically twice a year. If the need arises, "up to the minute" reports can be provided to the services in response to priority requests.

Table 4. lists the titles of the reports generated by the Footprint prototype utilizing the five specified data sources. The reports are grouped into three functional categories; Force

<u>FORCE STRUCTURE</u>
- Primary MOS vs Duty MOS
- Assignment Profile
- Gender Trends
- FY86 Year End Gender Profile
- FY86 Year End Force Structure
- FY86 Authorized Force Structure by ASI
- Authorized Quantities by Unit Identification Code
- Projected Authorized Assignment Profile
- Projected Force Structure Trends

<u>PERFORMANCE INDICATORS</u>
- Accession Quality Trends
- Accession ASVAB Trends
- Mental Category Trends
- FY86 Year End Mental Category Trends
- FY86 Year End Education Profile
- Years of Service Trends
- Retention Rate Trends
- FY85 Year End Retention Rates
- Qualifications for Initial Award

<u>TRAINING PROFILES</u>
- Quantity Trained by Training Type
- FY86 Training for MOS XXX
- Training Course Length
- Training Class Graduates
- Graduate Retention Rates
- Graduate Class Size

Table 4. Footprint Report Titles by Category

Structure, Training Profiles, and Performance Indicators. Each of the functional categories capture one or more aspects of the predecessor weapon system's MPT characteristics.

CONTENT OF THE FOOTPRINT REPORTS

A summary of the information contained in each of the 27 Footprint reports is provided below by report category.

Force Structure. These reports quantify the current and projected (required and authorized) composition of a specified MOS. The force is broken out by pay grade, skill level, ASI, Unit Identification Code (UIC), and/or Fiscal Year. Qualifications for initial award of the MOS area also included.

Training Profiles. The reports identify the One Station Unit Training (OSUT), Advanced Individual Training (AIT), as well as other training which has been or will be provided for a specified MOS and fiscal year. The number of enlisted personnel who previously graduated from a specified course and class is presented, along with the length of the class, the course attrition rate, and the percentage of class graduates who have stayed in the service subsequent to course completion.

Performance Indicators. Displayed are the historical trends, by fiscal year, of the population of a specified MOS. This includes mental category distributions, average aptitude scores, ASVAB score distributions (in the qualifying aptitude area), retention rates by pay grade, years of service trends, educational trends, duty location (CONUS versus OCONUS) trends, gender trends, and primary versus duty MOS distributions by pay grade. Further, the mental category distribution, average aptitude scores, ASVAB score distribution and gender trends are also displayed for all accessions of a specified MOS by fiscal year.

FOOTPRINT PROJECT STATUS

Footprint reports have been delivered to the U. S. Army Infantry School for the AAWS-M and AAWS-H, to the U. S. Army Signal Center for the MCOS, EOTF, and FHMUX, and to the Armored Family of Vehicles Task Force for the AFV. The reports were provided in two different formats, as complete sets of MOS data sets, and as complete sets of MOS reports grouped by system. The AFV Task Force has provided copies of the Footprint reports to it's three prime contractors and asked them to consider their appropriateness as the MPT baseline.

Subsequent to the deliveries, a Joint Working Group (JWG) was formed, comprised of representatives from the Army Office of the Deputy Chief of Staff Personnel (ODCSPER), Soldier Support Center - National Capital Region (SSC-NCR), and TPDC. Other organizations will be selected by ODCSPER, for representation on the JWG. One of the primary objectives of the group will be to work towards the institutionalization of Footprint within the Army as the Automated Target Audience Description Database. Additional areas approved for joint investigation are as follows:

The expansion of Footprint to the other areas required by the TAD guidelines, including anthropometric data, identification of high driver tasks, and performance data.

The expansion of Footprint to include Officer, Warrant Officer, Reserve, and Civilian personnel data.

The process by which the MPT data is "Up-Dated" throughout the acquisition process as more details are learned about the new system.

The best means of providing MPT data to Industry for use in designing the new systems within the specified MPT resource constraints.

POTENTIAL APPLICATIONS OF FOOTPRINT DATA

As the Footprint prototype effort began to produce reports and these were reviewed by Army data users, a number of potential applications were noted. Many of these will be used by the AFV Task Force and will support their decision process. The most significant of these are summarized below;

MOS to System Crosswalk. What MOS are associated with a particular weapon system or conversely, what are the various systems a specific MOS supports? (This capability will be provided by the Crosswalk project, which when completed, will serve as a front end to the Footprint).

MOS Restructuring. What is the composition of the present MOS, how are they distributed, and what kind of changes could be made to the organizational concept or force structure to reduce MPT requirements?

MOS Consolidation. When reviewing two or more MOS, what commonalities exist between them, what unique requirements can be eliminated, and what training packages can be consolidated?

Career Management Field Management. What is the status of the MOS relative to accessions, promotions, and

attritions? Which MOS are considered under populated or over populated relative to authorizations, requirements, and actual inventory.

Generic MOS Functions. When comparing multiple MOS with a common generic function (e.g., driving a vehicle), can training be consolidated, can manpower requirements be reduced, and can training locations be centralized to reduce training costs?

Weapon System - Manpower Trade-Offs. When summarizing the MPT of all MOS operating and maintaining a specified weapon system, can efficiencies of weapon system design eliminate undesired MOS requirements?

Distribution of Quality. When comparing the distribution of quality of a particular MOS, or group of MOS to the Army average, is an unequal distribution of personnel quality apparent? Is it significant?

Modification to Training Pipeline. When reviewing the training pipeline(s) associated with specific MOS or groups of MOS, can some high cost or lengthy courses be transitioned to on-the-job training, or embedded training, or can training be reduced through the use of job-aiding or expert systems?

Cross Service Data Exchange. Do comparable systems exist in the other services? What are their associated MPT profiles?

CONCLUSION

Results of the Footprint prototype have demonstrated that the integration of existing data serves a multiplicity of purposes that in most cases is quite different than those of the contributing data source. This synergistic effect enables the generation of a wide variety of MPT reports in a fraction of the time previously possible. It provides a historical perspective on various MPT facets which can be used to track MPT changes and reveal significant trends. It can serve as an automated means of modelling a vast number of variables in order to assess required trade-offs.

Initial discussions with services other than the Army indicate that not only is Footprint feasible for the other services, but that the burden of specific MPT requirements could be greatly reduced through the development of such a tool. Whether the tool is developed within each of the services or at a centralized data center such as TPDC, the resulting integrated data set presents a huge potential for identifying MPT constraints early on in the acquisition process.

But the Footprint project is not an end unto itself. The term Footprint originally referred to the MPT profile of an existing system. The Footprint project has demonstrated that MPT data can be aggregated by selected MOS or MOS associated with a particular system. In other words, the MPT reports can also be provided on any MOS at any point in the acquisition process whether they are associated with the predecessor system, comparable system, or new system. Current efforts are focusing on an integrated approach which will interactively support existing analytical techniques. This capability may one day provide an automated means of generating initial MPT reports, updating and modelling variations, and projecting MPT estimates.

Footprint is one small step for MPT, but one large step towards improving the Weapon System Acquisition Process.

ACKNOWLEDGMENTS

The Footprint concept was envisioned by Dr. G. Thomas Sicilia; Director of the Defense Training and Performance Data Center. The term "Footprint" was adopted as a label for the prototype effort.

The tools and methodologies developed would not have been possible without the guidance and foresight of the Soldier Support Center and the leadership of Major General Maurice O. Edmonds and Colonel Richard H. Terrell.

ABOUT THE AUTHORS

Major Joseph F. Fil Jr. is a distinguished military graduate of San Jose State University. He has served in Germany as a tank platoon leader, tank company executive officer, battalion adjutant, and battalion logistics officer. He recently completed an assignment in the Rapid Deployment Force(s), 24th Division, where he served as tank company commander and battalion operations officer. He is currently assigned to the Armored Family of Vehicles Task Force at Ft Eustis, Virginia, where he is responsible for Armor and Calvary systems and Manpower and Personnel Integration (MANPRINT).

Ms. Kris L. Hoffman is the Program Manager for the Footprint Project at the Defense Training and Performance Data Center (TPDC). She received an MS in Operations Research from the University of Central Florida and a BA

from Lawrence University. Prior to coming to TPDC she worked as an Operations Research Analyst for the Army's Project Manager for Training Devices (PM TRADE).

Mr. Don B. Johnson is a Program Analyst at TPDC. He is currently working on his thesis for an MS is Instructional Technology at the University of Central Florida. Prior to coming to TPDC he worked as an Education Specialist for the Naval Training Systems Center.

Mr. Dennis Collins is the Chief of the Personnel Plans Division at the Soldier Support Center - National Capital Region. He has an MA in Public Administration from Penn State. He previously worked as an Operations Research Analyst at the Army Infantry School on the close combat-light Mission Area Analysis (MAA).

AUTOMATED DEFICIENCY TRACKING

OR . . .

"OPEN" IS A FOUR-LETTER WORD

Ruth E. Roppe
Test Assistant
Deputy for Training Systems
Wright-Patterson AFB OH

ABSTRACT

During development, production and deployment of training systems, an efficient manner of tracking the deficiencies discovered during test and evaluation to their satisfactory closure is needed in order to provide a training system that will serve the user to the fullest extent possible. This paper covers the automated deficiency tracking system in effect and currently implemented in the Deputy for Training Systems at ASD. This system utilizes the Information Central (Infocen) mainframe system which has a data base management software system that handles fields and data--Battelle's Automated Searching & Indexing System (BASIS).

A real-time status is available during test by automating the deficiency data input in-plant as well as on site. Through the use of read-only capability, all users of the data are able to tie into the system via a personal computer, utilizing a read-only password, and look at their respective areas on a real-time basis.

INTRODUCTION AND BACKGROUND

As the responsible test organization (RTO) for the Deputy for Training Systems (YW), the Directorate of Test and Deployment (ASD/YWT) is the focal point within the deputate responsible for statusing/ reporting all aircrew and maintenance training equipment. Since a unique management situation exists in YW (i.e., we are our own RTO; there are multiple development test programs; test articles are software intensive; reported deficiencies average 1,950 for the first article tested), an efficient, workable way of tracking deficiencies found during test and evaluation was needed. The previous manual system was not responsive to our needs since it was not timely enough for daily use. The time lag in changing status and updating the system would put it at least a week behind. Validation, correction, recheck, resubmittal and closure all involved status changes that necessitated YWT notification to update the data base. Also, in YWT the personnel resources have been reduced significantly. Where once there were four test assistants to do the job, there now is only one position identified. This situation alone necessitated a change in the way test deficiency accounting had to be accomplished. The old system was accurate as far as the data entry was concerned; however, by the time the data was entered, it was outdated. In addition, the automated capability presented here saves much time and effort in many ways and by all parties. Since initial data entry is done on site, YWT is relieved of the massive input experienced at the end of test (especially the voluminous number of deficiencies identified on the first device tested). Also, the ASD test assistant does not have to compile a report for distribution giving the site an up-to-date printout of test progression and results, then have it printed (it is often necessary to photocopy and reduce it), and make out envelopes--all of which take time and resources. Often, because of the dynamics of the simulator deficiency resolution process, these reports are outdated by the time they reach their destination.

The management challenge was to develop a real-time system for tracking, statusing and reporting deficiencies identified during test.

SOLUTION

This challenge resulted in the development of the YW-26 system. The YW-26 system consists of an in-house management system (Infocen) responsive to our requirements, developed in conjunction with the ASD computer center, which satisfies the needs of the users.

Information Central (Infocen) is a division of the Information Systems and Technology Center (ISTC) of ASD. It is an outgrowth of a facility originating in the Air Force Avionics Laboratory in the late 1960s to provide rapid data processing of technical and fiscal information. Infocen tries to provide its users with s "one-stop" service shop for their data retrieval needs. This approach is "one-stop" in the sense that they try to provide the users with a wide variety of processing services (especially text processing) and guidance in areas where they are not permitted to act in the user's behalf (i.e., equipment procurement, etc.). They also provide information and technical assistance to help in preparing the justification package. With today's growth of on-line systems and the technical achievements in computer engineering, computers are becoming more and more complex. This further emphasizes the need for a "one-stop" shop arrangement to let the user concentrate on the management of his data, not on the technicsl aspects of computer automation.

The information processing power of Infocen has been made available to other Air Force and DoD organizations on a cost-reimbursable basis under the Federal Computer Resource Sharing Program. One of the primary goals of this arrangement is to reduce the overall costs of the computer services to any one organization. This supports the objectives of the Office of Management and Budget Circular-121 regarding cost reimbursable computer services (fee-for-service-basis).

The mission of Infocen is to provide management computing power to those organizations who need it but cannot afford a system of their own. One of the original problems was that the system was too expensive for any one organization to fund in its entirety. That resulted in an effort to make the product available to non-AFSC organizations to help share the costs since the system had the capability to accommodate more users. As a result, both AFSC and non-AFSC organizations have employed the system's capabilities over the years. Under the operation of the ASD Single Manager for Automation, this arrangement still exists today and at its inception provided the only known costs reimbursable, on-line text processing facility in the Department of Defense.

Over the years Infocen has been successful in supporting an increasing number of users and at the same time reducing the overall costs of the system. The sharing of the computer resources by many users has been a historical trademark of the Infocen system, and it is in full agreement with the government'a Federal Reaource Sharing Program. The bulk of Infocen activities has been and continues to revolve around a data base management system specializing in the management of textual information.

According to Col W. Figel, the current DCS for Communications and Computer Systems for ASD, "The number one goal of the Information Systems and Technology Center is customer support!" To do this, Infocen operates four Digital Equipment Corporation VAX 11/780 computers, two VAX 8700 computers (approximately five times faster than a VAX 780), and a large data base management system called BASIS (Figure 1).

One of the nicest features of BASIS is its "user-friendly" query language. In other words, it's easy to use. You don't have to be a programmer or systems analyst to figure out how to make BASIS do what you want it to do. Some training is required. But, fortunately, YWT has published the BASIS Read-Only User's Guide which helps a new user get started by the use of step-by-step instructiona. This document supplements Infocen's BASIS training classes and provides guidance which pertains to YWT's data base uae. Some of the features of BASIS besides its primary use for inquiry and retrieval purposes are:

a. It is accessible via remote compatible terminals which can call up the system 24 hours a day with world-wide operation.

b. It allows real-time access to data; i.e., atatistics and reports can be updated/generated at any time.

c. The data can be easily updated by simply diaplaying and inputting with the use of formatted CRT screens which represent actual hardcopy documents. This module of BASIS displays a form on your terminal ao you can "fill in the blanka" when entering data.

d. Documents can be word searched using key words or stems of words. BASIS has a "full-text" retrieval system which means you can search on any word, or any combination of words, without your search having to be pre-defined to the system. It lets you ask questions "on-the-fly" to meet your apecific and changing needs.

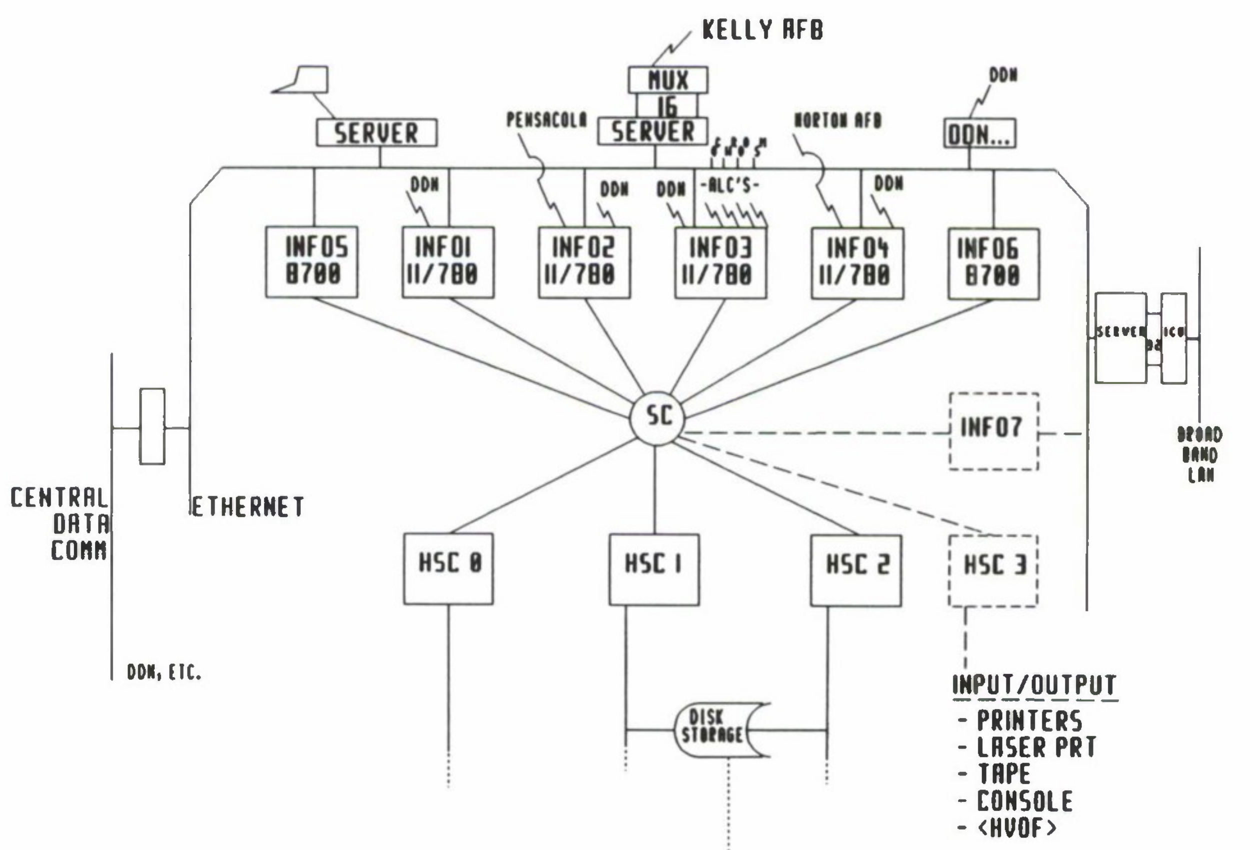


Figure 1

e. Batch processing makes easy execution of predetermined batch commands with a single command. With the Profile module, you can store a complicated search procedure and give it a name so that in the future you can repeat the procedure just by typing in the name.

f. The possibility of user-created reports is virtually unlimited.

Many DoD organizations use the system to track a wide variety of information such as:

* Aircraft and weapons systems deficiencies
* Status of scientific projects
* Formal military training courses
* Lead times for raw materials
* Office management reports
* Budget information and control
* Audio-visual products inventory
* Library documents control
* Logistics management systems
* Project management and milestone monitoring
* Contracts management
* Inventory control
* Legal systems
* Aircraft end items

HOW IT WORKS

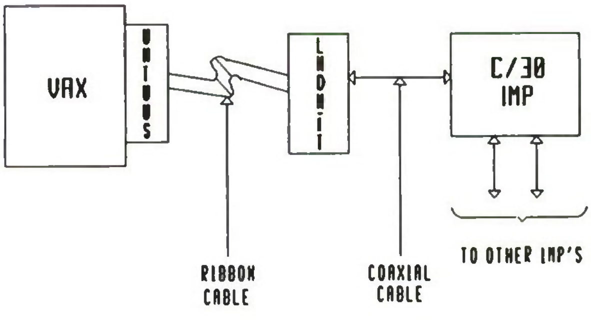
ASD'S SYSTEM

It became apparent early on that, in order to get deficiencies corrected in a timely manner, some type of statusing system was needed. ASD and the contractor not only needed to know if a deficiency was Open, Ready-for-Recheck (RFR), or Closed but also the specific stage in between Open, RFR, and Closed. From this dilemma emerged a series of status codes. Figure 2 identifies the various new status codes used by YW and depicts a flow chart which carries a deficiency from discovery all the way to closure at a Materiel Improvement Project Review Board (MIPRB). Figure 3 is the narrative for the flow chart of status code assignments. When the codes are clearly understood by all parties, the margin for misunderstanding who should be doing what, when, narrows dramatically. This in-depth tracking of deficiency correction more clearly defines for all players in whose court the next corrective action lies.

In order to data manage these deficiencies, the facilities of Infocen are employed with the use of BASIS which has been previously described. In BASIS the deficiencies are tracked by data fields. In using this system, the use of a smart terminal is recommended simply because of the many other benefits that can be utilized to better serve the user's needs.

The users of BASIS have the ability to tie in to the system regardless of their geographical location as long as they have a compatible terminal and telephone lines for long distance telephone transmission. There are three options for calling up the system--a commercial line; the Defense Data Network (DDN); or by use of a Local Host using DDN. The DDN is a computer network system set up by the Department of Defense that lets computers call "hosts" (customer-owned computers which are connected to a host port on an interface message processor or a store and forward packet switch. More generally, the host consists of the hardware and software components required to support end-to-end communication on the network.) communicate with each other. It was formerly known as ARPANET. The office of the Secretary of Defense on 10 March 1983 provided that "All DoD ADP systems and data networks requiring data communications services will be provided long-haul and area communications, interconnectivity, and the capability for interoperability by the DDN. Existing systems, systems being expanded and upgraded, and new ADP systems or data networks will become DDN subscribers. All such systems must be registered in the DDN's User Requirements Data Base (URDB)." The hardware required to connect the Infocen VAX systems and its configuration is illustrated in this diagram taken from the DDN User's Guide published by Infocen (Figure 4). Obviously, if there is the ability to tie in to DDN, long distance telephone expenses would be reduced.

The hardware required to connect the INFOCEN VAX systems contains three main components. The UNIBUS is the VAX-11/780 bus system for slower peripherals such as printers and terminals. The Local Host/Distant Host interface, or LHDH-11, connects the UNIBUS to the C/30 Interface Message Processor, or IMP. The UNIBUS and the LHDH-11 are connected by a ribbon cable. The LHDH-11 is connected to the C/30 IMP by a single coaxial cable. The local configuration is illustrated in the diagram below.



DDN Hardware Configuration

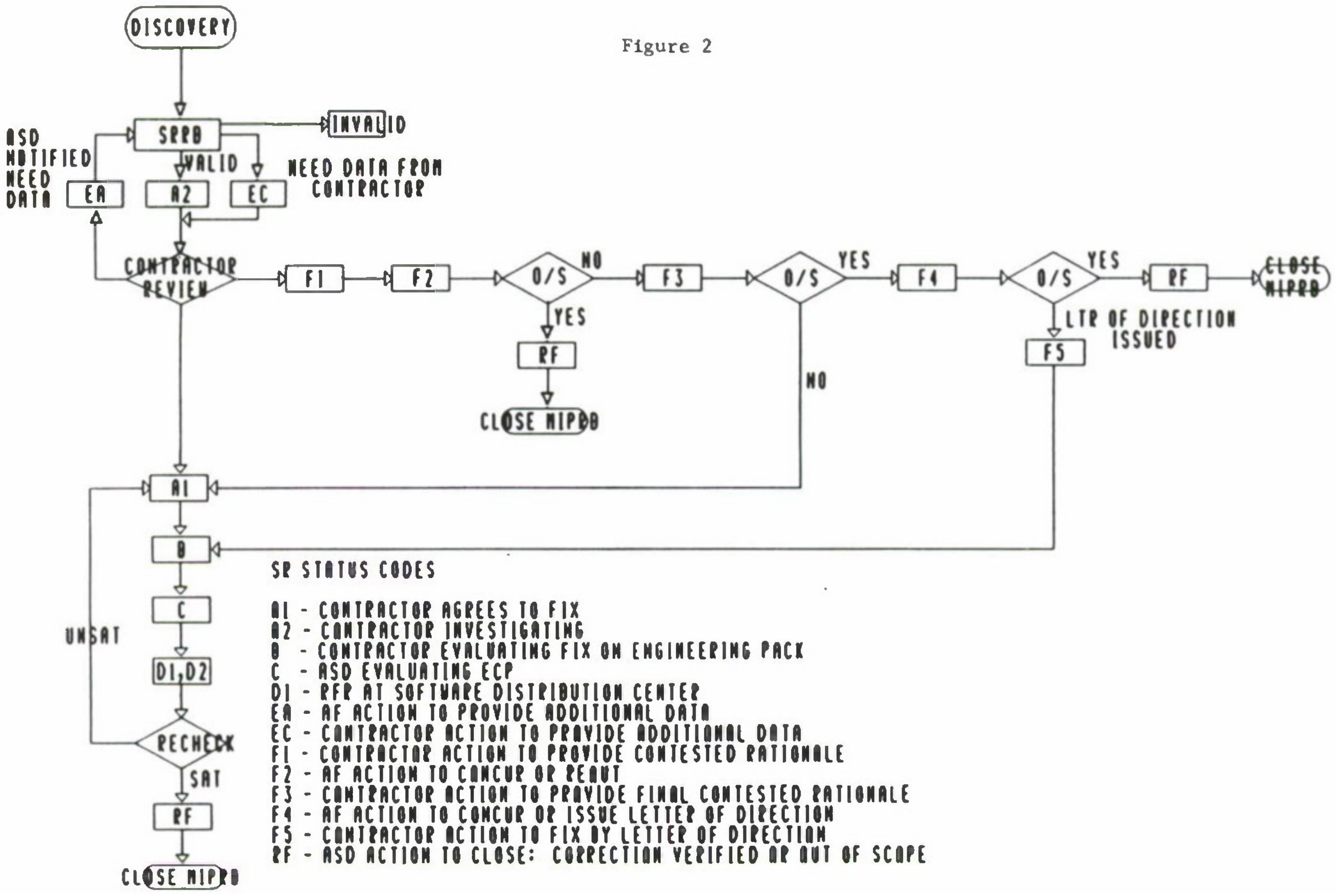
Figure 4

There are three major functions and two minor functions a user can perform on the DDN. The major functions include logging in to another computer, moving information from one computer to another, and sending electronic mail to users on other systems through the network. The minor functions let the site manager check the network status and monitor what DDN users are doing on his system.

For further information on the DDN and how to become a user, contact the Infocen Office (ASD/ACSP), Wright-Patterson AFB OH 45433-6503, (513) 255-6075 or AUTOVON 785-6175.

Potential users of the Infocen system and YWT's data base must request in writing to ASD/YWT whereupon a valid need is determined. If we feel the requestor's needs would be better served as a user

Figure 2



PROCEDURES FOR ASSIGNMENT OF STATUS CODES

1. Upon receipt of an SR, it is reviewed by the Service Report Review Board (SRRB) for validity. If valid, it is entered into our system, coded A2 and forwarded by YWK to the contractor. If validity cannot be determined, additional data can be requested from the contractor and the deficiency is coded EC.

2. Upon receipt by contractor, SR is restatused one of three ways: EA for cases when additional data is required from ASD; A1 for those SRs that the contractor agrees to correct; and F1 for SRs considered out of scope.

3. SRs that the contractor agrees to correct.

a. If the contractor determines SR is within scope, he agrees to correct the deficiency and notifies YWK of recommended status change to A1.

b. The status is next revised to reflect evaluation of correction action on engineering pack and is coded B.

c. Once corrective action has been finalized and the resultant ECP prepared and submitted, the SR status code is changed to C to reflect CCB processing and any remaining contractor design activity.

d. When SR is RFR, the code will next be changed to D1 or D2 indicating location of where recheck will be accomplished.

e. If recheck results are satisfactory, SR is recoded RF indicating ready for closure at the next Materiel Improvement Project Review Board (MIPRB). If results of recheck are unsatisfactory, SR is returned to contractor via YWK letter and recoded A1.

4. SRs considered out of scope:

a. If the contractor considers the SR is out of scope of the contract, he provides the contested rationale to the Air Force via YWK, and the SR is statused as F1.

b. SR is next coded F2 which reflects receipt of the contested SR from the contractor and Air Force reclama.

c. If we concur with the contractor's claim, the code is changed to RF, indicating ready for closure at the next MIPRB. If we nonconcur, an additional rationale for contested action is required, SR is coded F3, and it is returned via YWK letter.

d. If the contractor concurs with the Air Force that the SR is in scope, he agrees to correct the deficiency and statuses it as A1. Final contested rationale is forwarded if the contractor still considers the deficiency out of scope of the contract, and the SR is placed in the F4 category.

e. If we concur with the contractor's claim, the code is changed to RF, indicating ready for closure at the next MIPRB. If we nonconcur, the Air Force issues a letter of direction to the contractor and the status is recoded F5. The SR progresses through the cycle as the contractor evaluates the corrective action on the engineering pack (B to closure).

Figure 3

of our system, we must submit a letter to INFOCEN for a username.

After the username is assigned, the individual or organization then has the ability to log-on to BASIS at any time and look at whatever documents he has been allowed to see. YWT's BASIS User's Guide for Read-Only Capability gives easy-to-understand directions on how to log-on and use the system.

A real concern expressed at the inception of contractor read-only capability was maintaining the security of Infocen and YWT's data base in particular. This was alleviated with the following contractor log-on proceas. As a contractor logs on to ASD's system, he is first asked for his "username." (The username is established at the time a new user receivea clearance to log-on to Infocen.) When it clears, and the computer has determined the operator is, in fact a bonafide user, he must provide his user "password" which he alone selects and which it is recommended he change on a regular basis (usually monthly). The computer then queries him for the data base number of the data base he has been allowed to look into. In this case the data base number would allow him to look at documents created by the Training Systems SPO exclusively. When the computer compares the username with the list of users allowed to look at Training Systems SPO documents and finds a match, the contractor is allowed in to the data base and may proceed. He is now required to provide the data base ID which further narrows down his functions to read-only capability in that data base. Now, after passing all those singling-out obstacles, he must enter his own 6-digit contractor ID number, (provided to him at the time he received his username) which allows him access to documents with his unique USAF contract number found in the contract number field. Hia ID also limits his access to previously designated fields within those documents.

The use of read-only capability is a tool the using commands take advantage of aa well; however, unlike the contractors, they have the entire YWT data base to peruse. SAC has been taking advantage of this capability on the B-52 program for a number of yeara. When the MB-26 and B-1B programs become more TD/SR intensive, they will utilize the service to an even greater extent. TAC received their username in 1985 when a real-time knowledge of the deficiency status on the F-16 program became critical to resolution of Block 10/15 TDs. They also look into the data base for A-10 and F-15 facts. Around that same time USAFTAWC found that access to immediate deficiency data and status information on the F-16 program allowed them a better overall picture of the deficiency resolution process. Additionally, AFOTEC also uses the information on a real-time basis.

The Singer Company, AAI Corporation, and General Electric have had usernames for over a year. As previously noted, their ability to look into the data base is limited to documents and fields where they have a need to know. The usefulness of this capability is still being explored but seems to be expanding as it is utilized. Nearly four years ago a test assistant from YWT waa on site at General Electric, Daytona Beach, Florida, during in-plant testing entering test discrepancies as they were identified. These instantly became a part of the data base back at Wright-Patterson. At that time, if any member of the test team needed a count or printout of what TDs were ready for recheck or were in any other stage of correction, all that needed to be done was a little keyboard manipulation and a printout was available on site. If the test team had a write-up on a problem dealing with radar, for example, and they knew there had been previous write-ups on radar, all that was necessary to obtain a printout of all radar related TDs for review was to perform a word search on the writeups with radar in the detail portion. Had the GE in-plant entry not taken place, the TDs would have been written up on Test Discrepancy forma and held until the end of test. At that time the entire package would have been carried back to Dayton for data entry. Any tracking that needed to be done in plant would have been a matter of manual searching through cumbersome TD writeup books and TD logs.

On a "wish-list" for YW test personnel are lap-top computers which would be carried TDY by the test managers during test and used on-site for transmission of data back to ASD. Other test related duties and paperwork could also be performed with the computer's word processing capability. While the test manager is away from his desk back at the home office, correspondence could be created, then electronically mailed back to the office for coordination and dispatch.

There are many functions now available to the user with read-only capability, and the following are highlighted. (In keeping with its user-friendly reputation, more information about BASIS can be obtained by typing a question mark and the word COMMANDS. This will display a list of all BASIS commanda available for use.) A document or document set may be found and displayed. Or, if a printer is available, the information may be captured and printed for later use. The data retrieved may be sorted by any field, subfield, or partial field. It may also be sorted by multiple fields. As an example of multiple aorta, you might want all the deficiencies on a particular program aorted by device number and then the deficiency numbers sorted in ascending order (i.e., Item 1: B-52, WST2, TD#s 0007, 0053, 0392; Item 2: B-52, WST7, TD#s 0003, 0650). Reports may also be compiled allowing the user to select, arrange, label, and summarize information from the data baae and display or print the data as a report in columns with summary statistics, if any, listed at the end.

Since most users have a set of commands they use over and over to perform routine jobs, BASIS has a module called "Profile" which handles theae procedures. It will allow you to create, store, edit, and execute procedures, thus saving time and eliminating errors.

From the description above, it is easy to see that ASD's system is a very useful and user-friendly tool for manipulating the data input as the result of test and evaluation and tracking the deficiencies to their successful closure.

THE CONTRACTOR'S SYSTEM

The deficiency identification and reaolution process is totally computerized on the EF-111 program, the B-1B program, and the F-15E program. All new programs now have this BASIS compatability requirement at their inception as part of the Statement of Work.

The contractors use their own tracking systems utilizing whatever VT-100 compatible system they have established. They simply use a BASIS compatible type of aoftware program that can be output to a file. The mainframe system at ASD is a Digital Corporation VAX II/780. Configuration of the terminal is full duplex, ASCII, asynchronous, 300 or 1200 BAUD, 8 bit parity, 1 stop bit. (300 BAUD is

recommended for long distance phone lines.) ASD/YWT will work with the contractors to insure their system will interact with ours. The data can be sent directly over telephone lines; however, floppy disks can be hand-carried or expreas mailed back to ASD. But, clearly, this option sacrifices the timeliness of telephone lines. We need to be able to take this data and format it into BASIS format and store it on a tape in a BASIS file. We then load it into BASIS and scan it for errors, transferring it into our data base if and when we are satisfied it is correct. This allows real-time changes on line or off line.

Once the contractor has obtained a username, he can log-in to our system, enter the "MAIL" application, and send his data to us via commercial telephone lines or DDN. This operation can be done on a regular schedule (daily, weekly, etc.) or on demand. Once the system is working, it is incumbent upon ASD to get on line daily to see if there is data waiting to be downloaded. As soon as the log-in procedure is completed, a message appears indicating if there are any new mail messages (i.e., data waiting to be downloaded. The added capability of electronic mail makes communication with other users timely and cost effective, in addition to bypassing the cumbersome outbound mail process (correspondence and envelope preparation, and the time-consuming mail pick-up and delivery process itself) required of correspondence leaving an organization. Certainly this is not meant to, and should not, circumvent any contractual correspondence required to be signed out of the official contract offices at all levels.

As previously mentioned, a number of contractors are presently using the read-only system successfully and are able to look into the data base and see the latest status entered by ASD.

OTHER NOTABLE INFOCEN CAPABILITIES

Both the MAJCOMs and their operational sites find that the Infocen system with the read-only capability allows them more visibility and expedience in the process of acquiring a usable, fielded device to train on. Fortunately, the Air Force users of the system have access to a communications and terminal emulation program entitled "Call," whose software and manual have been licensed for United States military use only. Call can make your computer appear as a Digital Equipment Corporation (DEC) VT100 terminal to other computers.

Call emulates the ReGIS graphics command language used by DEC graphics terminals. Even when emulating non-graphics terminals, Call uses graphics to emulate the advanced text capabilities of the VT100 including 132 columns of text on screen at a time. It also has data capture. While you are communicating with another computer, Call records your conversation in memory or you can specify that this recording also be saved to a disk or printer.

Call has flexible file transfers. It supports simple text transfers using the protocols XON/XOFF and more complex text or data transfers using XMODEM. Call's command interface prompts the user with a list of available commands whenever he is asked to enter a command. This makes Call easy to use for first-time or casual users. Scripts of commands stored on disk to automate commonly used series of commands is another feature of Call. When you find yourself repeatedly entering the same sequence of commands, this feature becomes valuable. For example, every time you call and log-on to a particular remote system, you will probably use the same commands. Like an actor repeating the same lines every night from the script, Call can repeat the same sequence of commands over and over from a script stored in a file on your disk. Frequently used commands may also be stored in keyboard macros and assigned to any key. You create and modify scripts using any editor or word processor, as long as each command is followed by a carriage return and no special characters are embedded within the commands. Any Call command may be used in a script. Call only looks at the first letter of each command, so misspelled commands will be ignored as long as the first letter is correct. Scripts containing passwords should be protected to prevent passwords from being stolen. This protection may consist of only storing passwords on floppy disks and physically locking up the disks when not in use. Call will also accept commands from the computer with which you are communicating. These are remote commands. You as the user will never enter a remote command into Call; they are for use by system developers to make Call's operation more automatic. Remote commands may be used to reduce user interaction with two different computers.

In order to use Call, the United States military user would need the following:

a. A computer that works with Call (i.e., IBM PC, IBM compatible PC, Z-100).

b. An IBM compatible graphics adapter or a Z-100 (to use the graphics capabilities).

c. Another computer to talk to.

d. A data link via a direct line or modem.

e. The DOS operating system for your PC.

f. At least one floppy disk drive.

g. A printer (optional).

Also, since Call is not copy protected, it may be used with a hard disk drive.

CONCLUSION

The deficiency tracking system, as it pertains to software intensive items such as simulators, is a massive undertaking that requires automation as a means of keeping it under control. This undertaking must be a joint effort of the acquiring command, the using command, and the contractor; and it can and is being accomplished with the use of computers. With a data base of this size (over 44,700 documents) cooperation and communication are the keys to fielding devices that are suitable for the maximum training capability.

REFERENCES

Infocen Publication 103 - Everything You Always Wanted to Know About Infocen

Infocen Publication 106 - DDN User's Guide

YWT Publication - BASIS User's Guide for Read-Only Capability

YW Operating Instruction 66-1, Equipment Maintenance, Deficiency Reporting System, 24 Jan 86

Call User's Guide & Reference Manual, Written by Rich Blomseth, MicroSystems Co., 16987 Frank Avenue, Los Gatos, CA 95030, 16 Sep 85

TRAINING SYSTEMS LIFE CYCLE ENGINEERING CHANGE SUPPORT AT THE TRAINER SYSTEM SUPPORT ACTIVITY

Mr. Gerald J. Taylor
Naval Air Systems Command
APC 205-31
Washington, DC 20631-1205

Mr. Robert L. Stiegler
Naval Training Systems Center
Representative, Atlantic
Norfolk, VA 23511-5121

ABSTRACT

As the degree of sophistication of military weapon systems has increased, there has been a corresponding increase in the complexity of weapon systems training devices. The most explosive increases in complexity have occurred in those training devices which are software intensive. Increases in the amount and rate of change of weapons system software, coupled with increased trainer unique software, has resulted in inadequacies in trainer system configuration management and prime weapons system/training system concurrency. Many of the Naval Aviation front line aircrew and maintenance simulators have fallen one to three years behind the configuration of the weapon system they were intended to support. As a direct result, optimum utilization of these training systems has become difficult, eliminating or seriously reducing the Navy's ability to improve fleet combat readiness on prime weapon systems through training system use. After in depth analysis of practicable support options to solve these problems, the concept of providing on-site organic technical support for both the software and hardware of major weapons systems trainers was formalized. This program has been recently implemented for the AV-8B and F/A-18 programs. The on-site support organization is entitled the Trainer System Support Activity (TSSA). This paper will focus on the role of the TSSA in Engineering Change Support. This role includes providing: (1) a single point of contact on-site with technical knowledge of weapon system software/hardware as it relates to trainer systems; (2) rapid response to requests for trainer impact analysis, system engineering of proposed changes and cost-and-lead-time estimates; (3) timely design and installation of modifications to trainer systems; (4) trainer system configuration management and status accounting. To examine the TSSA role in context with the weapon system it supports, this paper will also describe the interface between the Navy activities supporting the aircraft weapon system software, those supporting only the trainer and the relationships that exist on the trainer site. The TSSA is located on site as is the Trainer Tactical Software Activity (TTSA). Also involved is the operational software controlling laboratory called the Weapon System Support Activity (WSSA) and the Naval Training Systems Center (NAVTRASYSCEN) with its subordinate Regional Offices. The involvement of the WSSA, NAVTRASYSCEN, TSSA, AND TTSA provides a flow of information that is essential to implement aircraft software changes into trainers in a timely manner. With the WSSA, TSSA and weapons system contractor operating together, the lag time between changes to the aircraft and the trainer is minimized and optimum utilization of the trainer is attainable.

BACKGROUND

Until recently operational software support by Naval Air Systems Command Field Activities of trainers for major aircraft programs was performed either through contract or by NAVTRASYSCEN's Cognizant Regional Offices (CRO). Lack of advance documentation often delayed contracting resulting in trainers that were too far behind the Weapons System to be efficient. The first learning steps to correct this lack of congruency problem between the Weapon System and the trainers were:

a. The P-3 Aircraft Program with NAVTRASYSCEN responsible for the trainer software support functions and NAVTRASYSCEN and contractor personnel, in coordination with the Naval Air Development Center (NADC) maintaining the software on-site at Moffett Field, CA.

b. The F-14 Aircraft Program with the Software Support Branch of the Pacific Missile Test Center (PMTC) Pt. Mugu, CA responsible for all software support functions utilizing a team of PMTC and contractor personnel to perform the functions.

c. The A-6 Aircraft Program with NAVTRASYSCEN responsible for trainer software support and software configuration control. Although the NAVTRASYSCEN field activities at NAS Whidbey Island, WA and NAS Oceana, VA were performing modifications, major modifications were contracted to industry firms.

d. The General Trainer Program with the NAVTRASYSCENREPLANT Pensacola

Division in conjunction with the NAVTRASYSCEN personnel at the Field Offices at NAS Memphis, TN and NTC Great Lakes, IL, providing trainer software and hardware modification support and configuration management utilizing NAVTRASYSCEN personnel.

Parallel evolution of weapon system and trainer software development and associated organizational functions is absolutely necessary if efficient utilization of trainers is to be attained. The lack of timely trainer software implementation and poor configuration management has contributed to problem areas which tend to degrade trainer effectiveness. These problem areas include lack of trainer currency, fidelity, availability, and increased Life Cycle Support costs as "catch up" becomes necessary.

THE PROGRAM

The Naval Air Systems Command Air Program Coordinator for Aviation Training Systems, APC205, has recently implemented a plan to establish on-site organic managerial and technical support for both the software and hardware of major weapons systems trainers. These on-site organizations are called Trainer System Support Activities (TSSAs), and are field activities of the Naval Training Systems Center. The NAVAIR Plan successfully brings together in an interactive working relationship two NAVAIRSYSCOM centers of trainer software expertise, the Naval Training Systems Center and the Weapons System Support Activities (WSSAs).

NAVTRASYSCEN has a long history of expertise in Training Device Operational Life Cycle Support. NAVTRASYSCEN engineers and their contractors are proficient in the development and modification of trainer-unique software and aircraft avionics/weapon system hardware and software simulation. NAVTRASYSCEN in-service engineers are collocated with the users of the trainers and are able to translate user requirements into trainer hardware and software modifications.

WSSAs such as Naval Weapons Center, China Lake, CA, Pacific Missile Test Center, Pt. Mugu, CA, and Naval Air Development Center, Warminster, PA, have developed teams of engineers and computer scientists, and contractors assigned to a Tactical Trainer Software Activity, (TTSA), who are very proficient in testing the hardware and software of the aircraft weapons systems. They have developed systems to simulate the combat missions of the aircraft in order to test and evaluate the weapon system hardware and software. Some of the hardware and software is common to trainers; some is simulated in the trainers. The WSSA personnel initiate and monitor engineering changes to the aircraft weapon system that require corresponding engineering changes to supporting trainers.

IMPLEMENTATION

With NAVAIR tasking to NAVTRASYSCEN and the WSSAs, the Trainer System Support Activities were structured and began coming on line in FY86 with the formal staffing of the AV-8B TSSA at MCAS Cherry Point and official recognition of the P-3C TSSA at Moffett Field. Figure I is a typical Trainer Support Organizational chart and is consistent with all the other weapon system trainer support organizations. The TSSAs are being established in two phases. Phase I will proceed through FY89 and includes the establishment of the F/A-18 activity at NAS Lemoore, CA, the E-2C at NAS Norfolk, VA, and the A-6E at NAS Whidbey Island, WA. Establishment of a TSSA for additional weapon systems platforms such as the F-14A/D and V-22A will be accomplished in Phase II and are scheduled to be implemented by FY89 and FY91 respectively. The full scheduled implementation is presented in Figure II and Figure III, with Table I providing a snapshot status of the TSSAs as of April 1987.

TSSA SITE	WEAPON SYSTEM	PHASE STATUS
* Cherry Point, NC	AV-8B	Functional
	KC-130	Introduction
* Lemoore, CA	F/A-18	Functional
* Norfolk, VA	E-2C	Functional
	SH-2F	Functional
	MH-53	Planning
* Moffett Field, CA	P-3	Functional
* Whidbey Island, CA	A-6E/F	Functional
	EA-6B	Introduction
* North Island, CA	SH-60B/F	Introduction
* Tustin, CA	CH-46	Planning
	CH-53	Planning
* Camp Pendleton, CA	AH-1T/W	Planning
* Kingsville, TX	T-45TS	Planning
* New River, NC	V-22A	Planning
* Miramar, CA	F-14A/D	Planning

TABLE I: SNAPSHOT STATUS

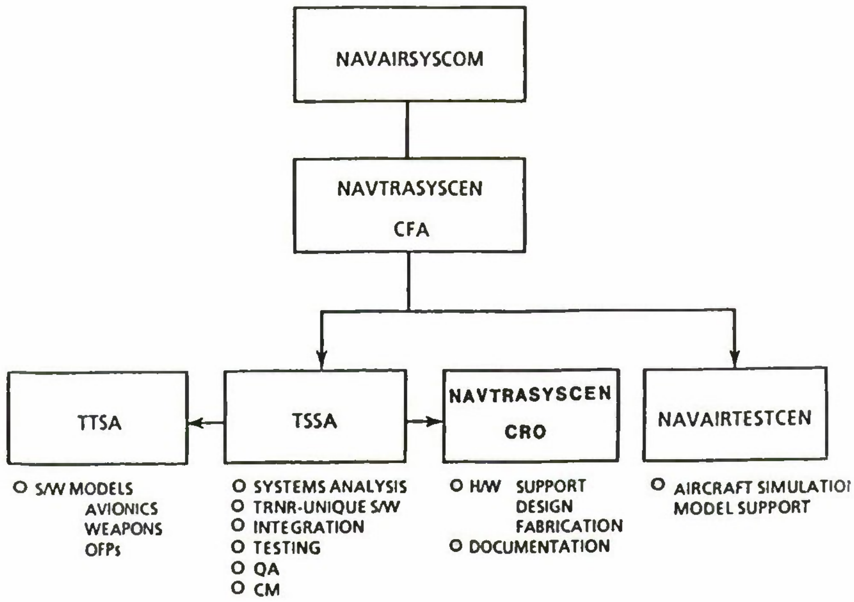


FIGURE I: TYPICAL TRAINER SUPPORT ORGANIZATION

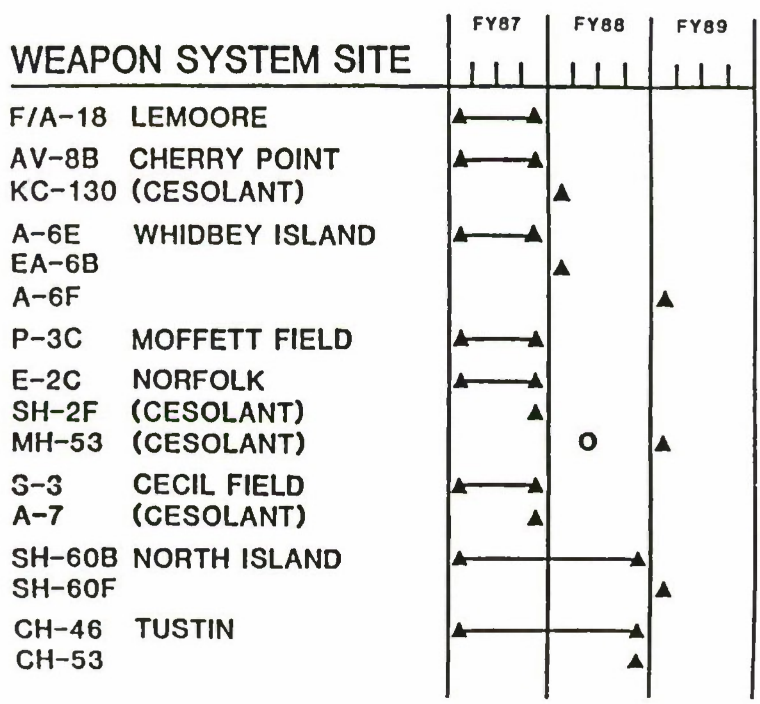


FIGURE II: PHASE 1 TRAINER SYSTEM SUPPORT ACTIVITY IMPLEMENTATION

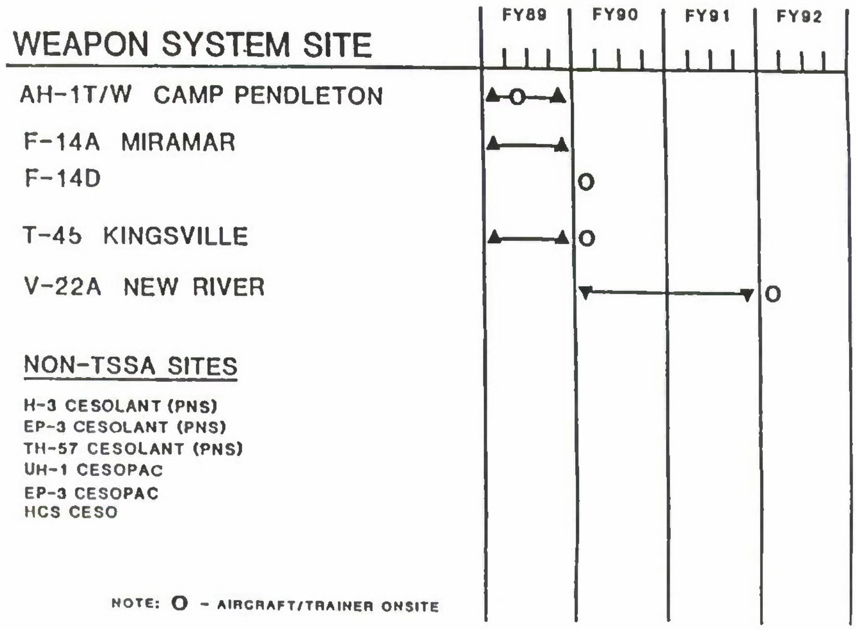


FIGURE III : PHASE 2 TRAINING SYSTEM SUPPORT ACTIVITY IMPLEMENTATION

TSSA PROGRAM

The Trainer System Support Activity is an on-site organic organization dedicated to trainer configuration management, trainer change support (modification), acquisition support, and fleet engineering support of weapon systems trainers. The on-site team consists of NAVTRASYSCEN engineers, computer scientists, and technicians, as well as contractor personnel. The contractor personnel work under their own supervision and perform modifications as assigned. The collocation of the TSSA/WSSA technical personnel and supporting computer equipment at the model manager/trainer site for the weapon system allows greater control of the modification program through direct interface with the functional wing commander. The TSSA is the focal point for weapons systems trainers in the field.

By designating the TSSA by weapons system type and collocating it with the trainers, it has been possible to baseline the software and hardware for the AV-8B and P-3C trainers and maintain these baselines at the trainer site. Definition and analysis of requirements, and the formulation of the corresponding cost and lead time estimates to implement them occur more efficiently because of the direct interface with the fleet and the tie-in to the WSSA. Once funded, the follow-on activities of designing, fabrication, testing, integration, and documentation of Quick Response Modifications (QRMs) are accomplished more expeditiously.

The TSSAs have absorbed the local In-Service Engineers (ISEs) and all associated resources and are the focal point for the acquisition support and operational life cycle support (including the Quality Assurance and Revalidation Program) of the trainers in addition to modifications. Because of the scope of this endeavor, additional resources are provided to the TSSA by the Consolidated Engineering Support Office (CESO) of the cognizant NAVTRASYSCEN Regional Office in the form of:

* Engineering Analyses and development of Cost and Lead Time Estimates for hardware modifications of TSSA Weapon System (WS) trainers.
* Designing of hardware modifications for TSSA WS trainers.
* Peak load software modification assistance for TSSA WS trainers.
* Configuration management of hardware of TSSA WS trainers.
* Hardware/software modification and configuration management for non-TSSA Weapon Systems Trainers
* Providing COTRs for hardware fabrication contracts.
* Contract Services
* Fiscal/Budget Services
* Facilities Engineering Services
* Prototype modification fabrication.

CONCURRENCY

The establishment of the TSSAs has already started to overcome the major problem in simulation of weapons systems - concurrency, i.e., the simultaneous updating of trainers and aircraft. This concurrent updating is particularly difficult when it involves a major modification or update that is of long lead time and has been contracted out. The flow diagram in Figure IV demonstrates the procedures now being successfully utilized to overcome these problems.

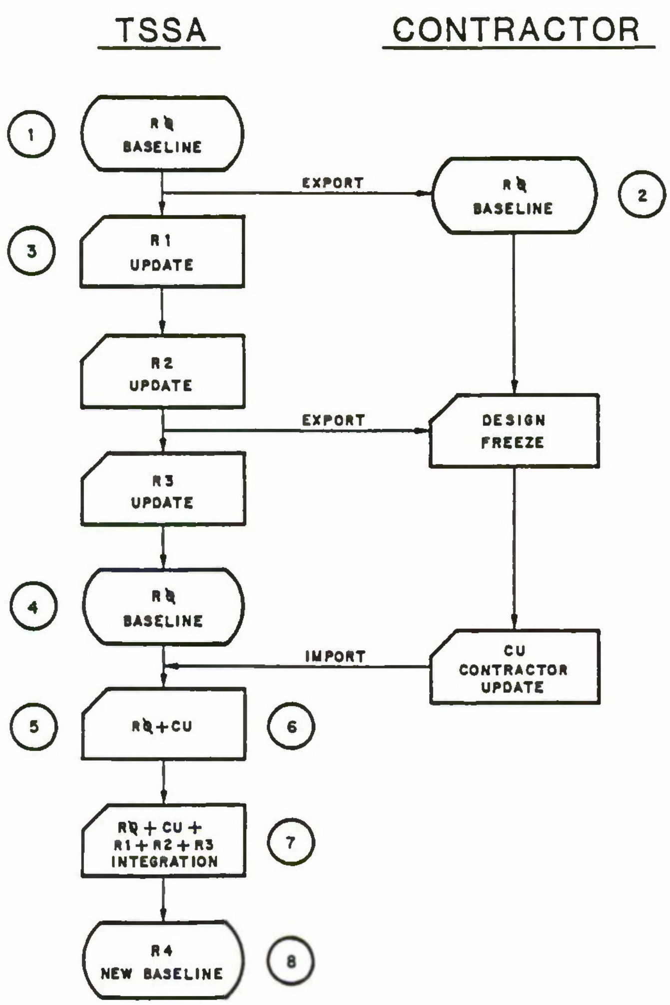


FIGURE IV : MODIFICATION FLOW DIAGRAM

1. A baseline (RØ) is established by the TSSA.

2. TSSA provides baseline to contractor who designs the contracted modification.

3. While the contractor is designing his modifications, the TSSA continues providing quick response modifications. In addition, design change information is provided to the contractor by the TSSA for design consideration/implementation prior to design freeze.

4. When the contractor is ready to install his modification, the RØ baseline is reinstalled into the trainer.

5. The contractor installs and tests his update.

6. The new program, RØ baseline plus contractor modifications is provided to the TSSA.

7. The TSSA integrates the RØ baseline plus contractor update program with the R1, R2, and R3 programs it has developed to form the new baseline (R4).

8. New baseline (R4) is established by the TSSA and the cycle repeats itself.

A similar process occurs with the development of new aircraft software by the WSSA. (See Figure V)

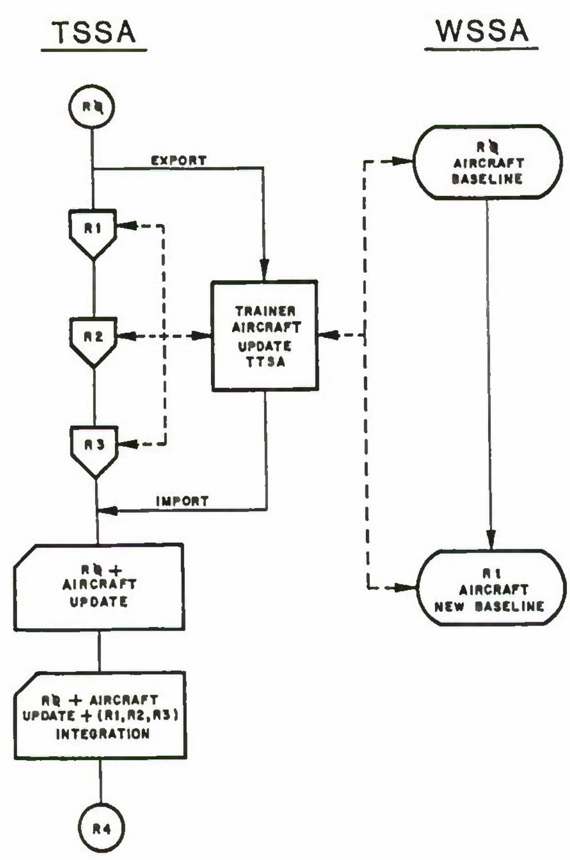


FIGURE V : TSSA/WSSA INTERFACE

The major advantage in this procedure is that the TSSA and WSSA are continually exchanging data and the two activities are continually interfacing through the TTSA. The WSSA is able to use the trainers to test some aircraft software and provide the TSSA with current progress of the effort and timely information so that the aircraft and trainer are simultaneously updated. Both of these procedures have been successfully accomplished on both the AV-8B and P-3C systems.

TRIAD

The success of concurrency depends on the triad relationship of the TSSA, WSSA, and Weapon Systems/Trainer contractor (Figure VI).

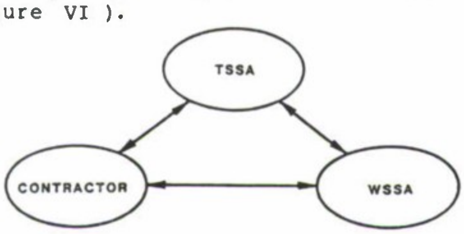


FIGURE VI : COMMUNICATION TRIAD

The Weapons System/Trainer manufacturer's involvement is not only that of incorporating changes in the weapon system platform, but also in providing relevant data and engineering interface between the three activities. The training and associated trainers are not separate entities from the aircraft but are an integral part of the parent weapon system, thereby requiring significant communications and exchange of pertinent reliable information between the activities.

MANAGEMENT

The key to the successful operation of the TSSA is the NAVTRASYSCEN site manager. This manager's responsibilities include:

* Configuration control, configuration management
* Supervision of trainer software development
* Coordination of In-Service Engineering
* Providing technical assistance to the Contracting Officers Technical Representative
* Analysis of impact of changes on trainer fidelity
* Coordination of "Tiger Team" efforts, including installation of software and problem solving at sites other than the model training site
* Chairmanship of the local hardware and software change control boards
* Supervision of trainer software system design
* Resource management
* Maintenance of the software baseline
* Interfacing with trainer users
* Being the focal point for the weapon system trainers (Figure VII).

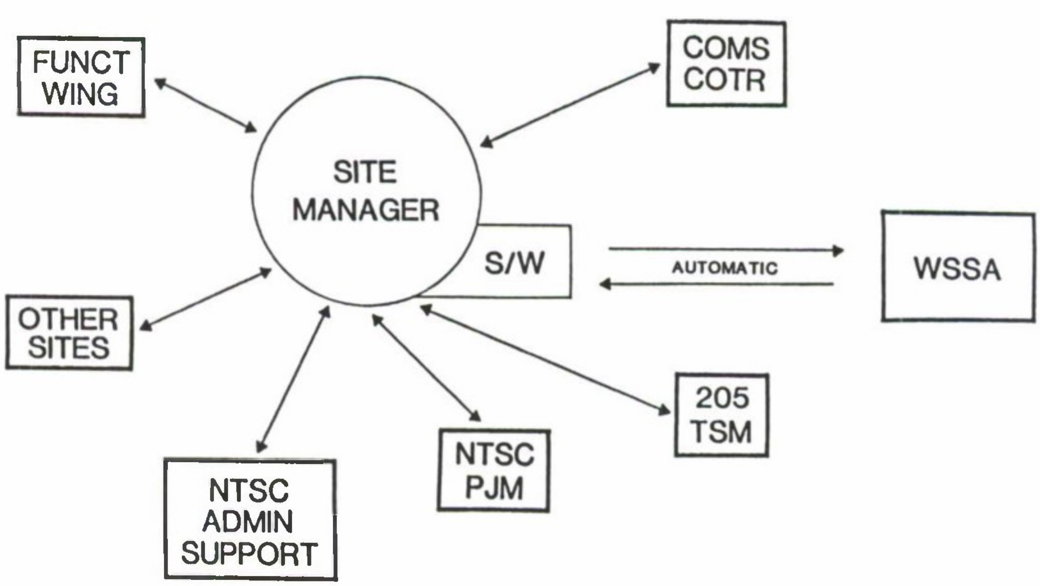


FIGURE VII : TSSA SITE MANAGER ROLE

RESULTS

Since its inception a short time ago, the P-3C TSSA and AV-8B TSSA have had remarkable success. The P-3C trainers are current with the airframe software, and the last update did in fact occur concurrently. The AV-8B TSSA has baselined the trainers, installed Omnibus III, upgraded the trainer operating system, and is prepared to install Omnibus IV as of this writing, again making the trainers concurrent with the aircraft.

The emerging TSSAs, with the proper support and management, are expected to have the same level of success.

SUMMARY

The TSSA concept has demonstrated a high level of performance and efficiency, with substantial documented cost savings. Continued effort in this area will provide enhanced training with a high degree of fidelity to the user activities throughout the life cycle of the weapon system and its associated trainers.

ABOUT THE AUTHORS

Mr. Gerald J. Taylor is the Assistant Deputy for Tactical Aircraft, Maintenance Systems Engineering in the Air Program Coordination Office for Training Systems, Naval Air Systems Command, Washington, DC. He was responsible for the planning and implementation of the Trainer Systems Support Activities and is currently responsible for the acquisition of TACAIR Maintenance Training Systems. He is a graduate of the Memphis State University (B.S.) and University of Arkansas (M.S.) Schools of Engineering. His experience includes 28 years in Naval Training Systems, 10 years as a Tradevman in the U.S. Navy, 15 years as a Senior Field Engineer for the Naval Training Systems Center, and 3 years at the Naval Air Systems Command in training system acquisition.

Mr. Robert L. Stiegler is the Head, Aviation/Land Field Engineering Division, Naval Training Systems Center Regional Office Atlantic, Norfolk, VA. He is responsible for the engineering, modification and operational logistics support of Aviation and Marine Corps Training Devices on the east coast. He is a graduate of the University of Wisconsin (B.S.) and Roosevelt University of Chicago (MBA). His experience includes 21 years in aviation and training systems, 6 years as an Radar/Avionics Technician in the U.S. Air Force and 15 years as an engineer at the Naval Training Systems Center.

BIBLIOGRAPHY

1. Chief of Naval Operations, "Life Cycle Management of Mission Critical Computer Resources (MCCR) for Navy Systems Managed Under the Research, Development, and Acquisition (RDA) Process," OPNAVINST 5200.2B, 25 Sep 1986.

2. NAVAIRSYSCOM "Evaluation of Policy and Guidelines for Navy Air Crew Trainer Software Support Activity Operations," CAA-84044-1-NAVY, March 1985.

3. Naval Air Systems Command, "Software Life Cycle Management Plan for AV-8B Aircraft Operator and Maintenance Trainers," REV. A, 17 Feb 1987.

4. Naval Training Systems Center, Training Systems Bulletin, P-1550-110, Summer/Fall 1986.

LIFE CYCLE SUPPORT FOR MARINE CORPS MULTIPURPOSE RANGE COMPLEXES - LESSONS LEARNED

Gordon Steven Dow
Naval Training Systems Center
Logistics Division (Code 4313)
Orlando, Florida 32813-7100

ABSTRACT

The U. S. Marine Corps is procuring two Multipurpose Range Complexes (MPRC) for combined arms, tank, and armor vehicle training. The need for both live and simulated fire capability ranges has increased dramatically due to a greater awareness of the need to train to standards, the costs of live ammunition, dangers inherent when using live ammunition during training, environmental restrictions, and the training limitations of non-automated ranges. The procurement of automated ranges was a new venture for the Marine Corps, with little in-house knowledge of such procurements. In addition, interservice agreements for joint procurement (to effect economies of scale) for similar training equipment required procuring the MPRC using multiple contractual vehicles (equipment procurement, range construction, ILS data and supplies procurement, and operation and maintenance life cycle support services procurement). The initial lack of definition of which parts of a range actually made up the training system added to confusion during this procurement effort. This paper documents lessons learned in logistics planning for the MPRC's. This includes the ILS elements which must be analyzed in the planning stage, unique ILS considerations discovered during procurement, and the operation and maintenance support strategy selected. The Marine Corps MPRC's are unique training systems with unusual (but not unsolvable) logistics support problems. This paper documents the joint approach taken by the Marine Corps and the NAVTRASYSCEN in solving these problems. This paper concludes that the modern automated range is experiencing a technical evolution and that the lessons learned in this procurement should be valuable to Marine Corps and Army range development and logistics personnel in the years ahead.

INTRODUCTION

Historically, the Marine Corps has used inventive, although comparatively primitive, methods for improving marksmanship through target practice. Some of these primitive target mechanisms, still in use today in remote areas such as Camp Hansen, Okinawa, include wire-run trip controls with counter-weighted "pop-up" targets and rather ingenious "moving targets" which employ the gravity force available from the gentle slope of a hill, and a remote controlled wire-trip release. Although these "automated" targets have a number of drawbacks, they seldom malfunction, logistics is not a problem, and their procurement costs are under $10 each with a lead time of just under 48 hours! On the other hand, they certainly lack the positive-feedback training features of the automated targets available today.

The target mechanisms available today represent state-of-the-art technology in both live-fire and simulated-fire training. The latest targets pop-up out of nowhere, they move, they disappear, they seem to shoot back (flash and bang), and even indicate when they have been hit. The latest technology includes accurate means of counting the number of "hits," providing the unit commander with actual kill scores made by unit personnel. As a result, they provide realistic threat presentations and objective measures of performance.

The Multipurpose Range Complexes (MPRC) represent the most sophisticated combined arms range target systems ever fielded. Equipment includes stationary and moving targets representing both infantry and armor threats. Supplemental equipment includes hostile-fire simulation (audio and visual, for both tank and small arms), and tank-kill simulators. The MPRC can be run fully automated by computer with previously generated scenarios, or all or part of the equipment can be controlled manually by an operator in the range control tower. Figure 1 illustrates the typical configuration of an MPRC.

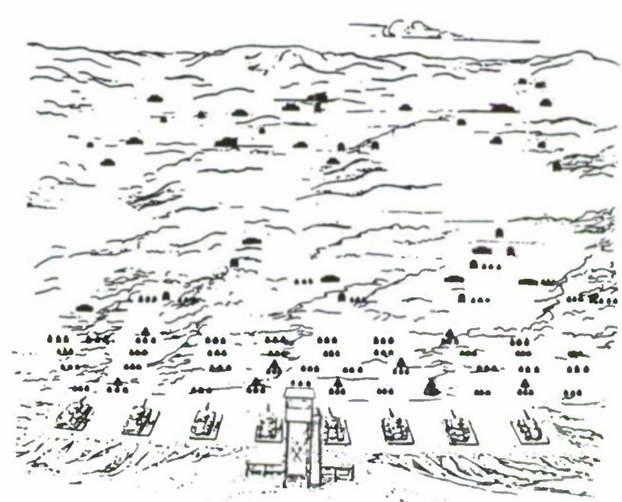


Figure 1: Typical MPRC Configuration

Initial credit for the procurement of this system goes to the Army. Procurement started as the acquisition of the Remote Targeted System (RETS), and the Armor Moving Target Carrier (AMTC). This evolved quickly into the Multipurpose Range Complex concept. Due to the existing interservice agreements for procurement, the Marine Corps frequently "piggy-backs" their procurements with those of the Army. This was the case with the MPRC. The acquisition of the two MPRC's for the Marine Corps (one each at MCAGCC, Twentynine Palms and MCB, Camp Pendleton,

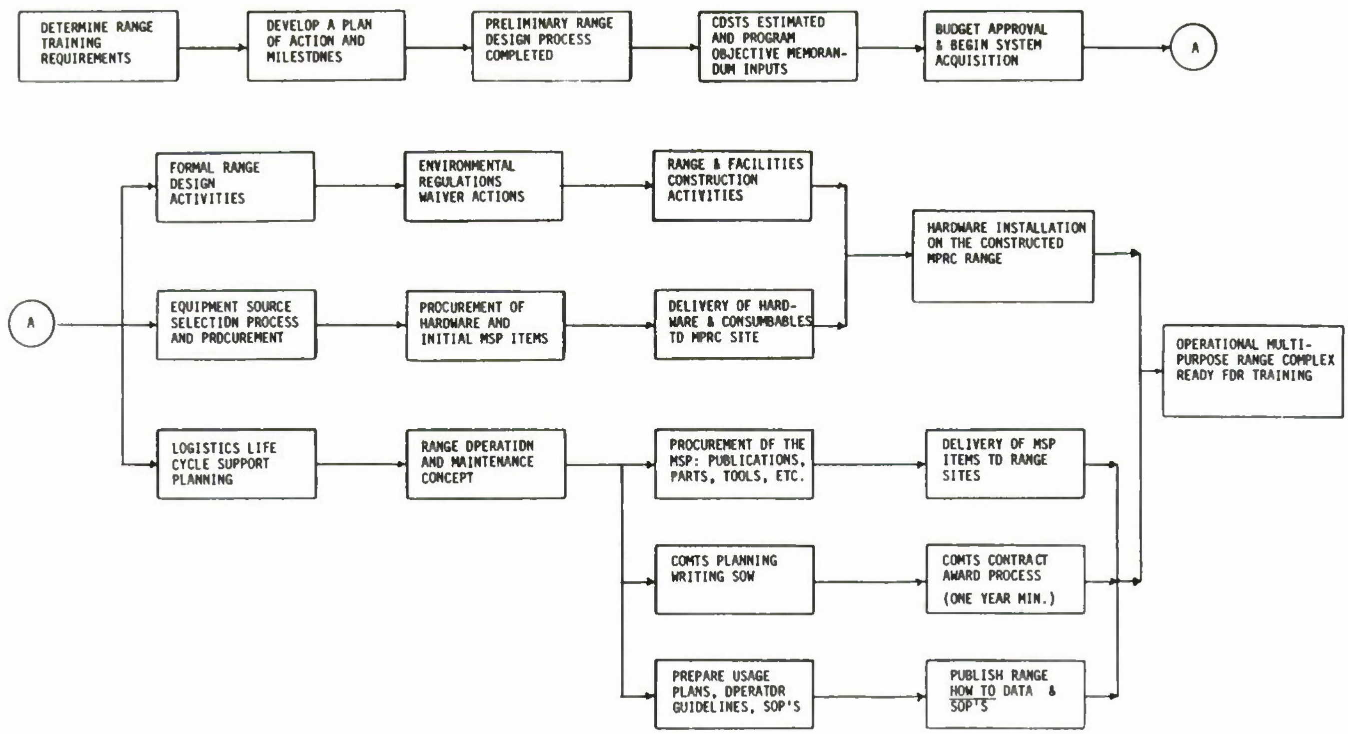


Figure 2: MPRC Procurement Flow Chart

both in California) has been managed by the Marine Corps Development and Education Command (MCDEC), Training, Audiovisual and Gaming Support (TAGS) Center, Quantico, Virginia. Headquarters, Marine Corps (Code TAP) has also played a role in this acquisition.

For life cycle support guidance and Contractor Operation and Maintenance of Training Systems (COMTS) procurement, the Marine Corps utilized the Naval Training Systems Center (NAVTRASYSCEN) in Orlando, Florida. The NAVTRASYSCEN has named the USMC-Ground contractor operation and maintenance program COMTS to distinguish it from the Navy COMS program developed for NAVAIRSYSCOM. This acronym splitting was done to avoid confusion when referring to the practices and precedents set by one program which may not automatically apply to the other program.

PROCUREMENT ENVIRONMENT

The MPRC acquisition was a complex undertaking. As shown in the accompanying flow chart (Figure 2), tracking, managing, and coordinating the many processes during acquisition is complex and relies on the efforts of a large interservice/contractor team. This includes an effort by the Corps of Engineers (Huntsville) to design and construct the ranges through a contract awarded by the Western Division, Naval Facilities Engineering Command (WESTNAVFACENGCOM); procurement of the MPRC hardware piggy-backed to an Army acquisition administered through AMCCOM, Rock Island; acquisition of additional logistics support items through the Naval Training Systems Center (NAVTRASYSCEN) and through other commands by procurements initiated by the Marine Corps through the MCDEC TAGS Center; and the acquisition of COMTS through the NAVTRASYSCEN. In addition, personnel at each MPRC site are responsible for some functions during the procurement process. Overseeing such a complex procurement, with the project team representing three service components and multiple contractors, creates a challenging environment for the project officer.

RANGES AS TRAINING SYSTEMS

The Old Concept of Ranges

Under the old concept of target ranges, an area of land was set aside where live-fire training could take place. This training could be for anything from handguns to the main guns on tanks. The targets were usually inert items, such as stacks of used tires, old hulks of vehicles, or other home-made targets. Such target shooting provided some training, but had limitations. Many times, the observer couldn't tell if a round hit the target and, to find out, it meant inspecting the target down range to see if any fresh holes were in it. Such a procedure could not be conveniently undertaken after every few rounds. Yet target feedback is highly relevant in a training environment. The lack of feedback limited the value of training received when using inert targets. It can be said that the old ranges were rather primitive, dangerous, and the positive training benefits were not easily measurable. This was doubly true without any accurate, safe, and efficient scoring-feedback system.

The New Concept of Ranges

The late 1940's witnessed training system development and the worldwide marketing of relatively sophisticated devices using electronics and hydraulics technology. Over the last 45 years, the technology for automated targets has advanced steadily. Probably the first significant improvement was the ability to reset the target for repeated use. The next major improvement was most likely some type of motion system. Along with these advances came control, making the target pop up or lay back down again if it had not been hit. Control systems gradually evolved from boxes with switches, to digital keypads, to computer controls. Target peripheral equipment was developed for added realism. These peripherals included hostile fire simulators, and target kill simulators for tanks, which included flashes and black smoke discharges. Such innovations added realism to training and improved effectiveness.

While these target improvements were being made, the Marine Corps was finding it increasingly difficult to fire live ordnance in their tanks. The reasons for this are both economical (each tank round fired costs thousands of dollars) and environmental (particularly because of complaints from nearby communities).

Fortunately, as nearby communities grew and ammunition costs rose, the laser beam was developed and with it the Multiple Integrated Laser Engagement System (MILES tm). By placing MILES sensors on automated targets, part of the training problem was solved. The tank gunners could fire harmless MILES lasers at the targets, and the targets would electronically determine (through coding) whether the incoming "round" could kill the target, and by impact position on the target, whether the "round" would kill the target. If the target was "killed," target kill simulators would automatically indicate this by flash and black smoke. In addition, the marksmanship training could now be scored and graded. Performance could be measured on the number of hits, where the target was hit, and the response time involved. And both the instructor or unit commander and the trainee would receive feedback as quickly as necessary to enhance the effectiveness of training with minimal costs and no environmental impact.

A New Definition of Ranges Is Required

As we see the ranges of today grow increasingly sophisticated, one thing is certain: A modern automated range is a major training system. We may choose to exclude it from the definition of one type of training system or another (making it Cog 2"O" or not), but it remains that such ranges are training systems of enormous significance to the Marine Corps and are here to stay. If the past can be used as a trend indicator, these ranges can only grow more complex and more sophisticated as technology advances.

One factor which originally made it difficult to accept these new ranges as conventional training systems is that the rangeland has become an integral part of the training system. As the sophistication of ranges continues to increase, the "integration" of the land into the training system (as a distinct part of it) is inevitable. Foxholes, tank trails, and access roads are cut into the rangeland. In addition, the land is sculptured to enhance target location and visibility and, by building berms, to protect otherwise vulnerable target mechanisms. Bunkers are often built right into the land, to house the most sophisticated tracked moving targets when they are not in use. The control and power cables are buried in the grounds of the range. In effect, the land becomes as much a part of the training system as the cabinet housing a classroom panel trainer. It can be seen then, that the land must be considered as part of the modern range training system; although historically, real estate has seldom been treated as part of a simulator or training system. Initially, this presented some difficulty due to various existing regulations whose authors obviously never considered the evolution of the modern range.

A related concern which has tended to keep ranges excluded from the conventional definition of training systems is that such ranges often include buildings which are part of the system. Paramount among these is the range control station (Figure 3). This is a tower which overlooks the range and houses the computerized control system.

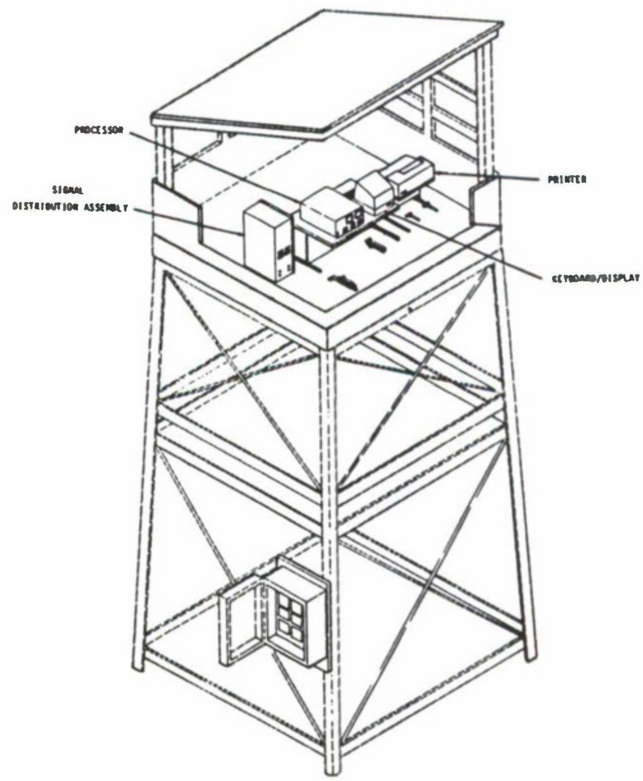


Figure 3: Range Control Station

In certain environments, it is critical to the operation of the MPRC that the tower's environmental stability be maintained within fairly close tolerances. The same applies to the protective bunkers for storing the armor moving target carriers (AMTC -- Figure 4). These AMTC's are quite costly and will experience greatly decreased Mean Time Between Failures (MTBF) when continuously exposed to the elements.

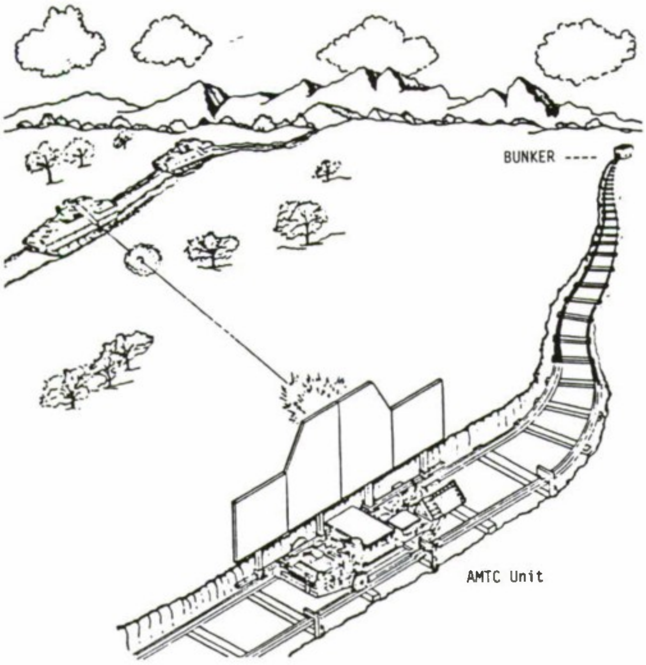


Figure 4: AMTC and Bunker

Both the terrain and certain buildings must be considered part of the MPRC training system. Historically, such properties have been the maintenance responsibility of Naval Facilities Engineering Command personnel at each Marine Corps Base, not the personnel who are tasked to maintain the training system equipment. Given this situation, it is a good idea to include in any COMTS-type contract at least those building and terrain responsibilities which are maintenance significant to the operation of the range. This was the strategy eventually selected at NAVTRASYSCEN for the MPRC COMTS contract and such a strategy can help offset some of the potential problems which arise during the operational life cycle. The project officer should bear in mind, however, that obtaining waivers to include such facilities maintenance in a COMTS contract is time consuming and requires diligent effort. Past experience has shown that this is time well spent, with the pay back returned as increased operational availability of the range training system, as well as potential savings from the award and administrative expenses of monitoring two or even more contracts which are critical to the range operation.

Examples of Problems Which Could Arise

The major problem which can arise when some facilities are not part of the COMTS contract is lost training. When a facilities responsibility is maintenance significant, the COMTS contractor has the right to request closing the range due to unsafe operational conditions (such as a broken window in the range control tower, malfunctioned air conditioning, building water leaks, etc.), until the base maintenance crews can correct the problem. This could take considerable time, especially at an MPRC site like Twentynine Palms where the MPRC is twenty miles from the base mainside. If a tank battalion is scheduled to use the range a certain week, and the range is closed because of a broken tower window or no air conditioning, considerable shuffling will be necessary to reschedule this battalion. In fact, the training opportunity may be lost altogether, which certainly has an adverse impact on Marine Corps readiness. On the other hand, if the COMTS contractor has the responsibility to fix this broken window or air conditioner and his monthly payments are contingent on operational availability, there will be little likelihood that training will be disrupted. Such potential situations were recognized early in the MPRC COMTS contracting effort, and actions were initiated to include this maintenance in the contractor's responsibilities.

A second type of problem arises from the same situation when facilities personnel respond to a trouble call. A deteriorated berm which requires reshaping is a typical example. If a government employee reshapes the berm, but in doing so damages delicate target lift mechanisms which are in immediate proximity, or severs a buried power or control cable, the Government is clearly responsible for this damage. If the COMTS contractor is required to repair the damage caused by the Government, this would be outside the scope of the contract and would cost additional money to the Government. The range would be closed until the damage was repaired. Once again, training would be disrupted and Marine Corps readiness would suffer.

No Easy Solution

The principal objective of contractor operation and maintenance services is to increase availability of training systems for training -- certainly not to substitute a new set of problems for an old set of problems. Making range availability contingent on a COMTS contractor's performance, which is contingent on base facilities personnel responsiveness to trouble calls, is unsatisfactory. It creates a daisy-chain of responsibilities with too many links capable of breaking down. The obvious solution is to include all functions which are maintenance significant in the COMTS contract. This may be an obvious solution, but it is not an easy one to implement.

To implement the concept of including maintenance- significant terrain and building maintenance, the project officer will have to use a multi-faceted approach to the problem. Bases may have differing policy and regulations regarding maintenance of buildings and land. Procurement policies

will vary from contracting one contracting activity to another, as well. The project officer must blend these differing rules, policies, and priorities to accomplish the goal of a COMTS contract which is more responsive to the needs of the Marine Corps.

Eventually, we may see modern ranges defined completely as training systems, including the land and buildings they occupy. Until then, each project officer should treat a modern range as a complete system, and provide life cycle support based on that premise. Until current regulations are changed to catch up with the realities of modern range maintenance, to accomplish this will require briefings and considerable correspondence to explain the need for waivers. It will also entail obtaining waivers and agreements from those agencies normally responsible for maintenance of such land and buildings. The project officer should allow considerable time for this, beginning early in the procurement.

LIFE CYCLE SUPPORT PLANNING

Logistics Planning: Part of the Procurement Process

In the past, many acquisition project officers were less accountable for, and knowledgable of, logistics requirements than they are today. This is particularly true in the Marine Corps, whose basic mission never has stressed the importance of long-term logistics support. Project officers are often faced with urgent turn-around requests for training support; they have to move quickly to acquire funding; and they are then faced with the nagging and expensive logistics support issue. If they include a logistics tail, it means an unacceptable delay in the procurement and/or more funding requirements than they have been able to beg, borrow, or transfer. All too often in the past, the Marine Corps has turned a blind eye to logistics support in order to field a training system. Once fielded, such systems have poor records of availability due to lack of logistics planning. With the emergence of support requirements as a necessary consideration, logistics is beginning to be understood and more emphasis is placed on training system supportability during acquisition.

As with the acquisition of any large system, front-end analysis (in the sense of analyzing and evaluating all aspects of the acquisition, not from the educational aspect only) and planning for an MPRC must include life-cycle support planning. One key to life-cycle planning is always the Maintenance Concept. The project officer needs to ask: "Who will operate and maintain this range? How will the person learn how to do it? What data, supplies, and tools will be needed to maintain the system to the level specified in the maintenance concept, to successfully provide the desired availability? What should that level of maintenance be?" These are basic logistics questions, but questions which may be overlooked by a young tank or infantry officer doing a first tour as a project officer.

Planning and budgeting for comprehensive operation and maintenance services in support of the ranges comes from the project officer's consideration of these logistics factors. Many of the requirements of a sound ILS Plan are covered in the data item description (DID) for INTEGRATED LOGISTICS SUPPORT PLANS (ILSP), UDI-L-23419A. This DID, or any of the many other ILSP DIDs which are very similar, can be used as a guide for the preparation of a Logistics Support Plan -- which the project officer can then use to ensure that important logistics elements are not overlooked.

Procuring the Material Support Package

An MPRC, like any other training system, requires a complete material support package (MSP) to provide the materials necessary for effective logistics support. The MSP is acquired based on the system maintenance concept discussed previously. The MSP is vital to the success of any COMTS contracting effort, since it includes the basic elements required for any maintenance effort. The MSP includes a complete set of technical data which reflects the intended maintenance concept. The MSP also includes all special tools and test equipment necessary to maintain the training system. The Marine Corps MPRC's are intended to be provisioned with an Accompanying Spare Parts Kit (ASPK). The custody of the ASPK will be transferred to the COMTS contractor at the beginning of the operation and maintenance contract, with the contractor using these parts to accomplish daily repairs while replacement parts are ordered to restock the on-site inventory. Depending on circumstances, parts could be replenished using the National Supply System or the COMTS contractor could order parts directly from the original equipment manufacturers.

The need for heavy equipment to perform terrain maintenance is an MPRC support requirement, a requirement which is not normally part of training system material support packages. It may not be necessary for the Government to purchase such equipment, but it <u>is</u> necessary to consider the requirement. The Base Facilities Department may provide the machines and manpower to maintain the range terrain, although this is perhaps the least responsive method of providing such support. If the operation and maintenance of the range is to be contracted, it will be necessary to identify this terrain maintenance requirement in the Statement of Work (SOW). The project officer then has the alternative option, if funds permit, of procuring the necessary heavy equipment as part of the training system tools and special equipment, or of establishing a requirement for the COMTS contractor to supply and operate the heavy equipment. The COMTS contractor then may provide the heavy equipment with his own operators or subcontract this responsibility.

COMTS Procurement

Once the logistics issues concerning the MSP have been resolved, the project officer must turn attention to operation and maintenance requirements. Recently, the Marine Corps has procured increasing numbers of target systems for their existing and planned ranges. These have varying degrees of operational and maintenance requirements. It is Marine Corps policy to employ smaller percentages of Marines in occupations which are not combat related. An example of this is the contracting of messing facilities at many bases. Maintaining targets could be done by civilians. Normally, it would be relatively simple to hire a few wage grade technicians to accomplish this. But over the last 25 years, the Government has instituted a policy of hiring fewer civil servants while encouraging the contracting of non-critical services. This policy relates to the guidance provided by OMB-Circular A-76, which spans four administrations -- proving it to be a durable bi-partisan policy. This policy is not likely to change in the foreseeable future.

With an environment of more complex training systems and limits on hiring maintenance personnel, the ground Marines had little choice but to look for a reliable alternative to operation and maintenance support. Since NAVTRASYSCEN was involved in planning, procuring, and the life cycle support of Navy and Marine Corps Cog 2"O" training systems -- and already had an existing functional and successful Contractor Operation and Maintenance of Simulators (COMS) program -- the ground Marines turned to the NAVTRASYSCEN in 1983 to begin development of a similar program for them. As stated previously, the Marine Corps program was given the acronym COMTS (Contractor Operation and Maintenance of Training Systems) to differentiate it from the U. S. Navy COMS program.

Most Navy COMS contracts have relatively straight-forward requirements. The Statement of Work (SOW) is the key document in all COMS contracts. It describes those operator and maintenance services to be provided by the COMS (or COMTS) contractor, and what the Government provides as a Material Support Package (MSP). The SOW also describes what functions continue to be the responsibility of the Government, and how the contractor's performance will be monitored.

In the case of the MPRC, the Marine Corps wanted a "turnkey" operation and maintenance contract. This includes a range operator, who operates the range during training periods; a range officer, who helps monitor the progress of the exercise and helps the Marines develop effective scenarios; security personnel (night watchmen), who help protect those ranges which are situated in fairly remote locations; a maintenance staff with skills in electronics, hydraulics, target fabrication, computers, and electrical distribution systems; logistics support personnel to maintain the supply levels of the on-site repair parts inventory and targets, and the baseline of the MSP. Certain economies of scale were obtainable by preparing a COMTS procurement package which would include several training systems (including the MPRC) at a specific Marine Corps Base, thus making a contractor skill-pool available for multiple training system maintenance responsibilities. This was intended to allow full use of personnel with specialized skills, such as hydraulics technicians.

Even though the Marine Corps desired a turnkey operation, some functions must be performed by Marines. These include, for example, ordnance handling for the tank gunfire simulators and hostile threat gunfire and target kill simulators as well as military police functions to apprehend any trespassers spotted by the COMTS contractor's night watchman. In addition, a Contracting Officer's Technical Representative (COTR) is required at each base to monitor and measure those elements of performance which are related to the schedule of deductions. A step-by-step analysis of all task requirements in the Operational Logistics Support Plan revealed these functions, and allowed for early planning. Without careful and detailed logistics planning, such embedded requirements would have been overlooked.

LESSONS LEARNED PROCURING THE MPRC'S

Front-End Analysis

In the sense used in this paper, front-end analysis is not limited to the educational aspects or training requirements studies exclusively, but refers to a much broader scope, including economic impacts, funding requirements, resource requirements, identifying as many steps of the procurement Plan of Action and Milestones Chart as possible, and so forth. It also takes a serious look at life cycle support, the maintenance plan and concept, and other ILS areas which contribute to life cycle performance and costs.

In the procurement of the Marine Corps MPRC's, the quantities of units and delivery dates were in a constant state of flux due to no overall front-end analysis and the unpredictability of funding. Front-end analysis was not accomplished because the procurement effort and the development of the MPRC concept ran concurrently. At the beginning of the procurement, when funding requirements were submitted, the MPRC concept did not exist. Instead, Armor Moving Target Carriers (AMTC - see Figure 4), Target Holding Mechanisms, Tank Gunnery (THM,TG - see Figure 5), and other target mechanisms were procured as individual units. The MPRC range, woven together as a training system, had not yet been envisioned. In addition, due to the DoD requirements for joint-procurements of like training systems, the Marine Corps found it necessary to merge their procurements with those of the Army. These factors limited attempts at front-end analysis. This situation made sound logistics and COMTS planning difficult, partly because the ILS requirements had to be revised every time

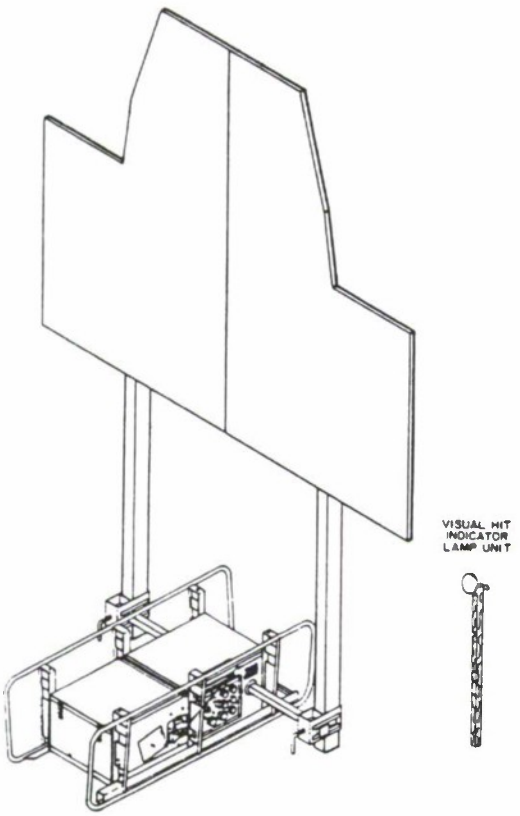


Figure 5: Target Holding Mechanism, Tank Gunnery

equipment was added, deleted, or substituted during the MPRC evolutionary process. Had the situation been different, front-end analysis would have helped in the planning process. Front-end analysis does not stop with looking at all the factors involved, but proceeds with the development of a plan that identifies all resource requirements, PMC, O&M,MC and MILCON funding requirements, as well as the needs for government manpower and time. Perhaps timing and coordination are the most important elements. The lesson learned is that a few months of planning can save years of delays as the procurement progresses. Detailed planning may not always be possible, as in this case, but every effort should be made in the future to ensure that a training system is fully defined before proceeding to procure it.

Multiple Procurements

It became apparent early in the MPRC procurement that multiple contracts would be required. This was due to DoD policy dealing with joint procurements of similar equipment. It is obvious that economies of scale may be obtained by combining interservice procurements into one contract, thus saving the Government various monetary and personnel resources. But it must be recognized that this policy also creates numerous management problems for the project officer when acquiring a system as complex as the MPRC. The lesson learned is that the project officer must be acutely aware of the many pitfalls which can result in such a multi-source procurement and make a special effort to coordinate, control, and manage the project given the difficult circumstances. Since, in this case, the project team included military and civilian personnel from three Armed Forces components, as well as contractor personnel, the project officer was faced with a difficult and challenging task. If future ranges are classified as training systems, and procured as a system rather than following such a tortuous procurement process, many of the problems encountered in the procurement of the MPRC's would be greatly reduced.

Communications

Another important lesson learned during the MPRC acquisition was the need for improved communications, although that can be said for almost any procurement. In this case, with the project team spread out across the country, and comprised of Marine Corps, Navy, Army, and Corps of Engineers personnel from multiple commands, as well as personnel from the equipment manufacturers, management, coordination, and communications were only slightly less complicated than untangling the Gordian knot. Frequent personnel changes added to the existing complexity. An effective countermeasure is and was to keep the team as small as possible and keep the lines of communications open. This could be accomplished by treating the MPRC (or similar project) as a complete training system and handle the acquisition as any "normal" simulator acquisition with the project team located at a single command. Another solution which could help alleviate this problem is for the project officer to produce a monthly "newsletter." This would serve as an accurate narrative file delivered to all team members and would include status of all project activity, a current list of all points of contact, and an updated milestone chart for each critical event in the procurement.

Maintenance-Significant Terrain and Facilities

One of the toughest problems in the life cycle support program was the critical issue of maintenance responsibilities for terrain and facilities which impacted MPRC operational readiness. A precedent has been set for the Twentynine Palms contract, but this issue may raise its head in the future. When a contractor is operating and maintaining a modern range training system for the military, it is essential that the range be considered as a training system, not a conglomeration of various targets strung out on a few acres of land. In the automated range, structures such as bunkers and control towers, and even some of the maintenance buildings, are a critical part of the training system. As range technology continues to evolve, the land itself will become increasingly more a part of the training system. We see it today with buried cables, terrain contouring, berms (which are critical to protect target mechanisms from damage), positioned

foxholes, drainage ditches, tank trails and access roads. If we keep in mind that improved Marine Corps combat training is the goal, and that the COMTS contractor's performance is contingent on the range operating as a unified training system, then including these essential maintenance elements as part of that system is critically important. The lesson has been learned, but the task still remains to define the modern range as a training system including the land and buildings it occupies. This may require changing some facilities and training system maintenance regulations whose authors never could have imagined the complexities of the modern range training system, nor the problems they created for those trying to provide sensible life cycle support.

Logistics Support

It should be recognized by now that the practice of procuring training systems for the Marine Corps without procuring a logistics ' support package must be eliminated. Although such a practice makes procurement of training equipment relatively easy for the acquisition project officer, it creates enormous support problems for the users and ultimately results is less effective training for the Marines for whom the equipment was procured. It is important to identify the agency responsible for logistics planning, procuring, and fielding. This should be done as early as possible in the procurement effort, to avoid the problems which have been identified in this paper. The solution, in the case of the MPRC, is to treat the modern range as a training system, and procure it with a fully staffed procurement team as is normally done for training systems. The logistics manager can then determine the MSP requirements, how this will be procured, who will provide it, and how verification will be accomplished. Identification early in the procurement process of who will take action to procure COMTS services is equally vital. The project officer must obtain cost estimates for the desired COMTS contract so that O&M,MC budgeting requirements may be identified and entered into the Program Objective Memorandum (POM). The logistics lesson learned is that the days are past when range equipment can be bought with serendipitous funds with the thought that logistics will somehow take care of itself. The equipment is too complex, and the logistics requirements are too extensive, to leave logistics planning to chance and costly ex post facto logistics procurement.

CONCLUSION

The lessons learned in the MPRC acquisition include: (1) Requiring front-end analysis and training system definition prior to contracting for equipment; (2) minimizing the adverse impact of multisource procurements (preferably by procuring the entire range as a single training system from one source); (3) improving communications among team members with diverse functions and distant geographic locations; (4) including land and building maintenance functions in the support contract for those items which impact the operational readiness of the range; and (5) including a logistics support package in all range procurement efforts. These lessons were learned at a cost of considerable time and effort to the initial MPRC procurement team. It will never be easy to procure a complex range training system. Many interdependent factors make the process like a differential equation: Change one variable and twenty others are affected, requiring continuous readjustments and realignments. But the lessons learned so far on the MPRC procurement can help future project officers bypass some of the vexing issues faced by the first project team.

In addition, modern ranges should be considered as training systems and not, as they once were, a conglomeration of targets strung out on a few acres of land. The training needs of today's Marine Corps are too critical to ignore this pressing evolutionary issue. Outdated facilities and training system maintenance regulations which never anticipated today's complex range training requirements should be revised so that they do not interfere with military training and readiness. If modern ranges were accepted as training systems, and procured as training systems with a standard procurement team, and if the issues of maintenance-significant terrain and facilities were resolved, most of the problems outlined in this paper would vanish. Modern ranges would still not be easy to procure, but they certainly would be easier.

The procurement of the MPRC was a complicated and challenging effort for the entire project team. It is still going on. As of August 1987, the contract for construction of the ranges is close to being awarded. The equipment acquisition is on-going. The Request For Technical Proposals (RFTP) for the MPRC COMTS contract (with other training systems at MCAGCC, Twentynine Palms, included) has been issued. It now looks like the MPRC's will be operational sometime in the first half of 1989. Many of the original MPRC team members have transferred or retired before completion of the project. It is hoped that the lessons learned pertaining to the Marine Corps MPRC, presented herein, will help future project officers procure modern ranges which are supportable and meet the needs of the Marine Corps during the entire expected life cycle.

REFERENCES

Request For Technical Proposal for COMS for MPRC Ranges et al, N61339-87-R-2028, including COMTS Statement of Work for the MPRC, N61339-452-MC-003, NAVTRASYSCEN Code 4322, Orlando, FL 32813-7100.

Operational Logistics Support Plan for MPRC's, NTSC-4313-USMC-OLSP-MPRC dated 15 August 1986 (draft), NAVTRASYSCEN Code 4313, Orlando, FL 32813-7100.

Data Item Description for ILSP for training systems, UDI-L-23419A, and others, available from DoD technical information centers.

Federal Acquisition Regulations, Part 37 - Service Contracting

Preparation of the Statement of Work, MIL-HDBK-245 (EC) dtd 1 Aug 77

Air Force Regulation AFR 400-28 BASE LEVEL SERVICE CONTRACTS, Vol I (C 1) dtd 13 Apr 82, Ch 2 Job Analysis.

OMB-Circular No. A-76 (Revised) dtd 4 Aug 83 - Performance of Commercial Activities.

OFPP Pamphlet # 4 dtd OCT 80, Supplement # 2 to OMB A-76 (A Guide for Writing and Administering Performance Statements of Work for Service Contracts.

ABOUT THE AUTHOR

Mr. GORDON STEVEN DOW is an Integrated Logistics Support Manager at the Naval Training Systems Center, where he has worked for 8 years. His work has been dedicated to USMC-Ground projects for the past four years, including the creation of the USMC-Ground COMTS program in 1983-84. Mr. Dow was a member of the original NAVTRASYSCEN COMS Group which was tasked to replace the disestablished TRADEVMAN rating with commercial contractors. He holds a BS (physical sciences) from the University of Central Florida, cum laude, and is a member of Tau Beta Pi and Phi Kappa Phi honor societies, and the Society of Logistics Engineers. Mr. Dow served as a Navy Electronics Technician during the Vietnam Era. As a published writer for the past 14 years, he has written five books and three hundred articles for national magazines.

RFP REQUIREMENT DEFINITIONS FOR CLS AND COMMERCIAL DESIGN

Robert Buss

Boeing Military Airplane Company
Simulation and Training Systems
Huntsville, Alabama

ABSTRACT

In recent years, there has been a growing trend toward including Contractor Logistics Support (CLS) options and commercial design requirements in acquisition contracts. Both of these changes are being implemented to provide life cycle cost savings. However, both changes are defined in Request for Proposals (RFPs) using previous military requirements. This paper addresses potential additional cost savings concepts using alternate requirement definitions to accomplish the same tasks.

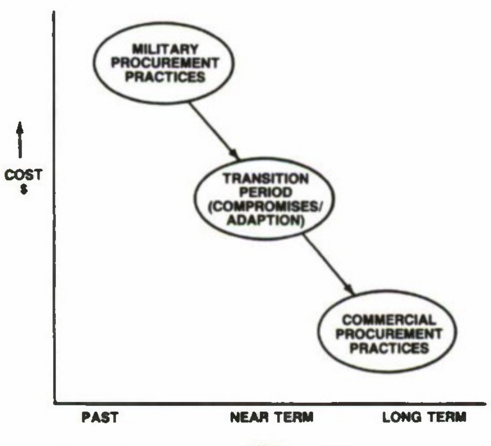
INTRODUCTION

There is an interesting Catch 22 that has developed with increasing technology. As weapons become more sophisticated battle-time decisions and engagement times becomes shorter. Therefore, the difference in who wins and loses is the one who is better trained. The Catch 22? As weapons become more sophisticated, they become more costly. This usually results in lower funding levels for supporting training programs.

To obtain better training with less dollars, significant concept changes are required. A first broad step in this direction has been evolving by use of Contractor Logistics Support and some commercial practices. This step was taken because it was generally agreed that most airlines acquire training devices and operate/maintain the systems at a much lower cost than the Government does for comparable units/tasks. But, what really causes the cost difference and have they all been implemented? This paper addresses that trend (Figure 1) in terms of three areas:

- Mission
- Military Culture
- Procurement Methods

FIGURE 1 MILITARY PROCUREMENT TREND



MISSION

One's first thought is that commercial and military missions are completely different. Although they have many common goals (operate at maximum cost efficiency, etc.), the real differences are in two major areas; when the training result will be used and currency to the weapon system.

Training

Commercial training leads to direct job application on a scheduled, predictable basis. Military training must prepare the aircrew to expect any segment or tactic relevant to a mission to ensure success. Therefore, military training is sometimes perceived to require more capability to try to better simulate that seldom-accomplished, potential task.

On a flight simulator, the most expensive subsystem is the visual system. A state-of-the-art visual system might cost $6 - $10M (excluding non-recurring). A night/dusk system will cost under $1M. Motion systems are relatively inexpensive ($400-$500K), but are facility drivers (high ceilings, strong floor, separate hydraulics room, etc.). Usually these decisions are already made in the RFP via the specification.

An alternative approach would be to provide the master task listings to the contractor and require him to propose the system based on his training analysis. In parallel, the Government would perform a study and drive the proposal to what is desired prior to BAFO thru the deficiency correction process. The difference under this concept is that the Government has the creativeness of industry to provide alternative approaches. If the result is the same, the Government decision is validated. If the result is different, cost savings benefits may be available.

Currency

Commercial aircraft configurations seldom change during the operating life. Military weapons systems constantly change in correlation to threat changes. This

requires a more flexible training device design and establishes a driver towards data requisition requirements.

Software is already as "flexible" as you can obtain. True, the update efficiency can be improved by more standard language usage and more disciplined design approaches but, software design is continually moving in that direction. Where then can improvements be made? Answer, in the "real world" simulations.

- A commercial route has infrequently changing airport stops. Therefore, it makes sense to present the actual airports on the visual display. A military pilot can go anywhere; but, it is not economically feasible to present all possible air fields. Yet, when military simulators are delivered, they usually have real-world runways for databases. A better answer might be to have the training people define database models based on training tasks to be accomplished in generic world terms (overwater approach, etc.). Each model would be specifically defined by drawings to allow visual system makers to develop common versions. Savings? -- Buy these databases only once for a given visual system.

- The visual databases could be taken one step further in that the data could be described in tables on tapes. Each visual manufacturer could (translate) the standard tape into his particular system. This approach would add savings where several similar models at a manufacturer exist, or for updates.

Similar approaches could be used for modeling of terrain, radar, ground station, etc. Is it more important to learn to land at Kadena AFB at night or a generic overwater approach in the dark?

Simulator hardware design already extensively uses commercial subsystems (computers, motion systems, visuals, etc.) in those areas which seldom change with aircraft updates. The area of update impact is in the replication of the aircraft panels/systems with secondary effects in linkage/cables. Use of actual aircraft panels/light plates and either actual or simulated aircraft instruments in the training device design is the significant cost driver in hardware updates. Savings are not easy to obtain here, but:

- In complicated systems where changes occur largely in PROM's, implement training device "hooks" in the aircraft design. These "hooks" will allow freezes and malfunctions to be accomplished.

- Establish a requirement in the aircraft prime contract to provide the aircraft common hardware to the training device manufacturer (obtains with aircraft priority and provides buy quantity savings).

MILITARY CULTURE

Every business, company and agency operates to philosophies which establishes its own unique culture. The result of this culture is inherent in the product. For example, an ideal product would be known for its high-quality performance (operating characteristics, reliability, etc.) and low cost. While the particular product (model) can change rapidly, the culture that produces it is slow to change.

Military culture is molded by discipline and the fact that it is an arm of the Government. The resulting proliferation of rules/regulations provides a rigid and rather inflexible culture. The product goal is high quality, but is manpower-intensive and expensive to accomplish. The point is that to fully implement a lower cost product (i.e. CLS), culture change is necessary and will take great efforts over long periods of time to achieve.

PROCUREMENT METHODS

Military procurement follows essentially the same steps to buy a space station or cockpit procedures trainer. Coupled with a perceived need for reprocurement capability and second sourcing, costs greatly exceed commercial approaches.

One commercial procurement approach (Figure 2) is simple:

- Define the aircraft configuration to be simulated.

- Provide a simple SOW defining trainer-user selectable features (motion/no motion, number of instructor CRTs, number of visual channels, etc.) and data needs (manuals, etc.).

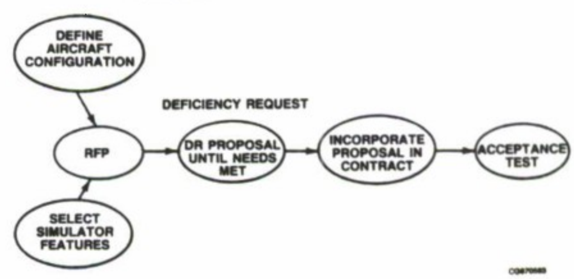
- Request an FFP-recommended price list of spares and support equipment (select by change order or incorporate in proposal).

- Review proposal and have modified (deficiency correction request) until proposal meets needs.

- Incorporate proposal into the contract.

- Accept the training device by one crew testing in the supplier's facility. Subjectively tune to that crew (total time about 1 month).

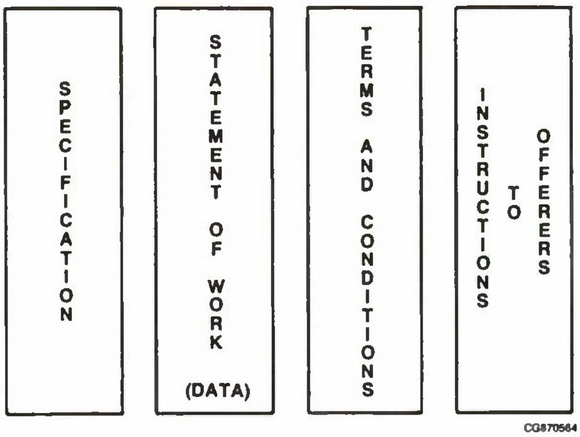
FIGURE 2 COMMERCIAL TRAINING DEVICE ACQUISITION PROCESS



This procurement approach provides a low-cost, proposal-request process, makes the supplier perform to his proposal promises and yields the desired device. Contractors save proposal dollars by submitting existing documentation describing their subsystems (linkage, visuals, etc.).

In contrast, the military procurement is complex, expensive, contains the "culture" requirements, and seems to be based on a low-trust approach. The remaining paragraphs in this section will discuss these requirements by each RFP (Figure 3) section (see Reference 4 for related information) followed by CLS comments.

FIGURE 3 RFP SECTIONS



RFP specifications are usually very definitive in design and test requirements. In fact, their problem is that they are too definitive. A good example is specifying brightness, resolution, etc. for a visual system that is an off-the-shelf item. On the surface this would not seem to be a big cost-driver because the system selected will meet the needs. But, the problem is the cost associated with showing how requirements are met at PDR, CDR, and repetition of qualification tests. The latter is done quite often even on Unit 453. Qualification by similarity? Doesn't happen -- it is harder to prepare and explain that analysis than to repeat the tests. And, how about those MIL Specs/Standards (included by the dozens)? The point is that most of the simulator systems (linkage, motion, visuals, etc.) delivered are commercial hardware with lots of dollars spent to prove they meet the specification. Each of these subsystems have their own commercial specification. Why not as a minimum allow them to be submitted with the proposal with equivalency proof? Once established, future inputs would only have to be simple updates. Also, when they are accepted as commercial off-the-shelf, testing should only be to prove that the items have been fabricated and installed correctly (acceptance-type testing).

Government testing is a significant cost and schedule driver. Typically SVT can take 2-3 months, DT&E 2-4 months, IOT&E 1-2 months; all followed by shorter repeats at site after installation. The first key to cost reduction is to recognize that, commercially, FAA requires data traceable to flight test results (Phases II & III). Again, levy the data requirement in the prime aircraft contract. With a solid design criteria base and the refined simulation models of today, a short check and subjective tuning period is possible. Subjective tuning is still required to fill the gaps particularly in areas where accurate flight test data is difficult to obtain (aerial refueling, etc.). Finally, during CLS, add requirements to solve problems in the real training environment. This can be "fenced" by X-Lines of code or Y-dollars or Z-person level of effort.

Once a good design criteria database has been established, how much testing is really required to ensure the product is satisfactory? Several team concepts have been proposed to reduce costs (See Reference 1). The one recommended in this paper is a compromise between the commercial and team approach. A small (2 or 3 persons) Government team would "don" contractor hats and participate during HSI and SVT. The benefits are both ways. The contractor has access to user crew knowledge. The Government gets to influence small changes in the most economical phase and provide early identification of major problems. This process would be followed by a short acceptance test at the site.

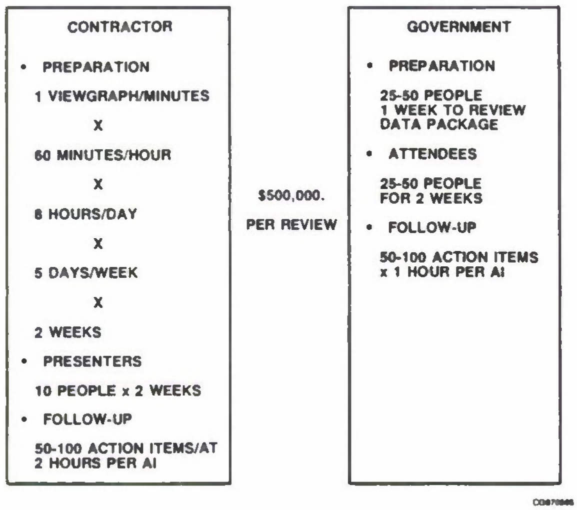
Reliability demonstrations when coupled with CLS have great redundancies. First a reliability demonstration must be passed, that only proves it worked over a very short time period. Then, after DD250, reliability growth tests are added. Lastly, not meeting availability guarantees imposes severe penalties and requirements to fix any pattern failures at no change in price. The recommended solution is to establish student thruput (not availability) guarantees with a dollar penalty for failure to meet thruput, but also provide strong incentives for exceeding the target (use positive profit motivation).

Statements-of-work generally define meetings, processes (standards) and data items. The most costly of these meetings are PDR & CDR. Contractor preparation of presentations is added work and for a typical major review can take several <u>thousand</u> hours. Coupled with contractor presentation time, Government attendance time and Government travel costs, total costs could approach $500,000 (Figure 4). A less costly method that would achieve the same result is to have periodic Engineering Design Reviews (say every 3 months) attended by a small Government team (5-8 people). Actual engineering data would be reviewed (drawings, etc. - not viewgraphs) with redline agreements made.

Data has many different areas of interest. First, how many data items produced are not part of the normal contractor process? If not part of this process, are they really needed? Assuming the contractor is doing initial CLS, it would appear that

the real driver should be to obtain sufficient data to support recompetition. Two types of data are needed; design and maintenance.

FIGURE 4 MAJOR REVIEW COSTS



Design data encompasses both hardware drawings and software documents. The first difficult decision appears to be format. To obtain lower costs using commercial equipment, a conflict exists on contractor processes called-out on the drawing. The solution is to include a top-level substitution drawing which cross references vendor standards to military standards. This yields enough information to create a modification drawing which is sufficient for incorporation of ECP's under recompetition. It does not provide reprocurement capability (only solution to that need is $).

Drawing media has become more difficult with increasing use of CAD. Datasets on magnetic tapes are probably not compatible with the new contractor's CADs system. The best bet for now is to obtain magnetic tapes (just in case) plus reproducible hard copies.

Software documentation has received considerable attention in the past few years. Many contracts now require updates every two months. This continual submittal requirement is expensive because of the cost of formal releases. A more economical method would be to maintain a redline set in notebooks which can be sample audited at the EDR's. This process would be continued during CLS with formal release updates scheduled periodically at points economically advantageous to the program.

Maintenance data acquisition in the past was strongly driven toward military technical orders. Currently, there is a shift towards acquiring commercial manuals recognizing the different environment of CLS. This trend should be strengthened with the added requirement for contractor's to either deliver the technician training course or guarantee Government access to the training course.

At the start of this data section, the question was asked about data generated by contractors versus contract requirements. A subjective answer is 50%. Many data items provide information as though the entire training device was developed from scratch. For example, the contract may require an off-the-shelf unmodified general purpose computer or realistic fidelity by high usage of actual aircraft parts. But the R & M safety hazard reports, etc. still require the breakdowns/analyses even though you can't change the design. The point again is to define the real requirement, put penalties/incentives in where they really count (at the end) and audit the contractor's existing process.

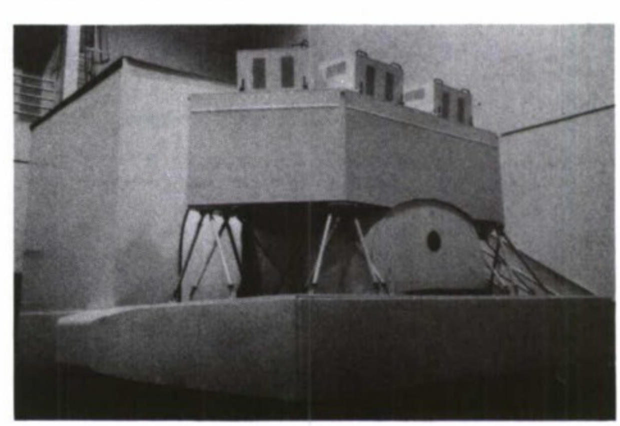
Terms and conditions largely influence cost on the "pain" principle. An example is Quality Assurance standards. Most contractors who meet MIL-Q-9858A recognize that it requires, for example, more explanation on purchase order flowdowns under MIL-Q-9858A than say MIL-I-45208A.

This added coordination or pain of doing business, requires added effort and thus costs. Commercial products continue to evolve toward better total quality to be competitive.

Instructions to offerers (ITO) included in RFP's are usually the combined result of several contributors and therefore tend to be somewhat repetitive and disjointed. A simple improvement is to make the ITO paragraph, the SOW paragraph and the WBS numbers all identical. The technical proposal then describes how one SOW paragraph will be met. The cost proposal automatically segregates correlating costs at the level stipulated.

Contractor Logistics Support requirements are going through the evolutionary stage. The biggest risks are inadequate manning, insufficient lay-in of spares and not enough replenishment dollars. During competitions, contractors try to get these areas down to the bare bones because they are cost drivers. For example, on the KC-135 Trainer System Program (MB-26 refurbishment - Figure 5) the difference in one technician was astounding (one technician per site x 17 sites x 2000 MH per year x 10 years = 340,000 MH).

FIGURE 5 KC-135 TRAINER



Barebones manning can be accomplished now by use of experienced ex-military personnel. But, in the future, this source will dry up and contractors will have to start their own training programs. RFP's need to increase requirements for manning/ spares rationale and establish better evaluation criteria. Additional recommendations are defined in Figure 6.

SUMMARY

The cost of a typical first-unit Government simulator is double the cost of a commercial simulator. Considerable cost savings can be achieved by streamlining the requisition process toward commercial acquisition techniques. Both commercial techniques and interim compromise recommendations have been defined in this paper. While some of these suggestions may on the surface appear unachievable, the following "can do's" are offered:

1. Redline drawings and software documents have been successfully used on the KC-135R Operational Flight Trainer program (Figure 7); for two years -- average availability 99%.

2. At last year's conference, Dennis Mathews reported on the outstanding success of an E-3A training program (Figure 8) acquisition.(See Reference 3) Some of its features were:

   a. No specification (only limited configuration description in SOW).

   b. Task Listings attached to SOW -- gave information on what training device had to do, not how to design it.

   c. No PDR or CDR -- EDR's every 3 months.

   d. No development or acceptance testing -- after RFT, a one year validation period required updates to achieve the required training.

   e. Practically no CDRL deliveries -- all required data for recompetition was put in a separate CLIN and priced for a future exercise date.

   f. Use of commercial training techniques--footprint reduced 34%.

   g. Accomplished modification of two airplanes, fabrication of a full-flight simulator and extensive modification of the facility -- all in 13 months.

This shows that the techniques exist that can result in significant savings and in some cases, improved results. Over the next few years, Government manpower and dollar constraints make this change critical to obtain not the same, but more and better training.

Can the culture change to allow these techniques be implemented? Yes. In closing, the following excerpts are

FIGURE 6 OTHER CLS RECOMMENDATIONS

Recommendations	Reason
• Include generic base support agreement as part of RFP.	• Difficulty in negotiating with each base
• Define base regulations requirements in base support agreement include security	• Not clearly understood at RFP stage.
• Better define site services/equipment available -- why can't the PMEL be used on a cost reimbursement basis?	• Schedule and spares pipeline quantities.
• Reduce QA practices during CLS to best commercial practice - alternative have contractor submit plan/incorporate as part of contract (See Reference 2 for other concepts)	• Significant cost driver that does not change performance.
• Use one budget source (color of money) to buy all CLS items.	• Allows better flexibility in mixing/ matching.
• Allow purchase of CLS consumables as part of spares lay in.	• Some contracts put in CLS CLIN - can't get in time to perform CLS then.
• Use Government sources to repair GFP.	• Difficult handling/sourcing prob.
• Allow acceptance of spares (DD-250) based on supplier certificate of conformance.	• Avoid expensive testing.
• Establish dollar limits on tracking/obtaining consumption (replenishment) data.	• Cost to obtain without benefits.
• Share other Contractor resources in same building.	• Avoid duplication of Xerox, fax machines and typing services, etc.
• Include a priced bucket to provide user required changes (instructor pages, etc.) - authorize via a site working group.	• Quick fixes to problems.

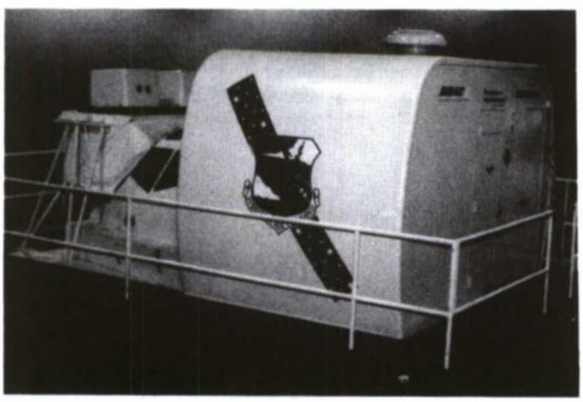
provided from a 1987 Under Secretary of Defense memorandum (See Reference 5):

1. "The DoD acquisition process is controlled by too many detailed, complex laws and regulations."

2. "I would like to test procurement methods more in line with commercial practices for both commercial and non-commercial products and services."

3. "The goal is to make it easier and quicker for contracting personnel to get line managers and commanders the quality products and services they need, when they need them."

FIGURE 7 BOEING KC-135 OFT LOCATED AT CASTLE AFB

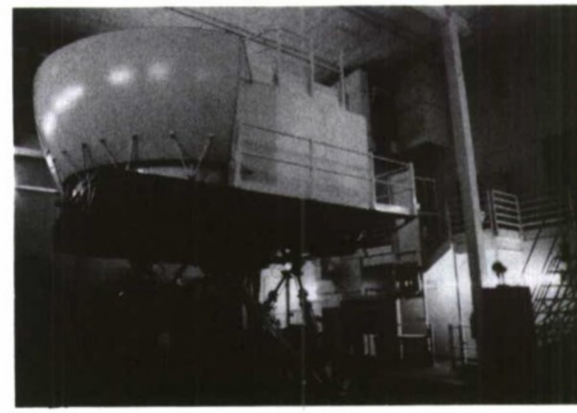


REFERENCES

1. Holzer, T. and Lynch, P., "What Ever Happened to Teamwork? A concept of Future Customer Acceptance Testing", Proceedings of the 8th Interservice/Industry Training System Conference, November 1986, pp 206-213.

2. Tierney, T., "Commercial Quality Standards for Training System Subcontractors and Vendors", Proceedings

FIGURE 8 E-3A FCT LOCATED AT TINKER AFB



of the 8th Interservice/Industry Training Systems Conference, November 1986, pp 224-226.

3. Matthews, D., "Practical Applications of Total Contract Training (TCT): Lessons learned from the E-3 TCT", Proceedings of the 8th Interservice/Industry Training Systems Conference, November 1986, pp 231-234.

4. Casperson, J. and Buss, R., "Proposal Management Through the Contractor's Knothole", Proceedings of the 8th Interservice/Industry Training Systems Conference, November 1986, pp 334-339.

5. Under Secretary of Defense Memorandum (A&L/I), "Procurement Regulatory Reform Test", 11 March 1986.

ABOUT THE AUTHOR

Mr. Buss is the KC-135 Training System Program Manager for Boeing's Simulation and Training Systems (S&TS) Organization. He has eleven years of simulation experience with half being in Proposal Management. Mr. Buss holds a BSNE Degree.

A-6F/F-14D AIRCREW TRAINER SUITE
RESULTS OF "BUYING THROUGH THE PRIME"

Robert E. MacLeod, M/S B64-01
Grumman Electronics Systems Division
Bethpage, N.Y. 11714

William M. Jarrott
NAVAIRSYSCOM, APC205-1A
Jefferson Davis Highway
Washington, D.C. 20361

ABSTRACT

The topic of the acquisition of Aircrew Training Devices through the Weapon System Prime Contractor ("Buy Through the Prime") has been addressed frequently in the past few years. The intent of this paper is to provide an objective analysis of this concept by utilizing the Navy's acquisition of the A-6F/F-14D Aircrew Trainer Suite (ATS) through Grumman as a case study. The paper will present an integrated Navy/Grumman assessment of the acquisition planning process starting with the Training Systems Requirements Analysis, proceeding through the Specification/Procurement Package Development, Competitive Solicitation, and ATS Program implementation. Special emphasis will be placed on the primary program goals of achieving delivery of the ATS concurrent with the Fleet introduction of A-6F and F-14D aircraft, as well as maximizing hardware and software commonality throughout the ATS Program. Other significant acquisition issues relative to Weapon System Contractor Furnished Equipment, Weapon System technical data, pre-planned configuration updates, and Integrated Logistics Support will be addressed in detail. The paper will compare the ATS Program progress to date versus both the initial acquisition plan as developed by Navy/Grumman and a projected acquisition plan that would have resulted if the ATS Program was being implemented by conventional non buy through the prime techniques. The analysis will provide "Lessons Learned" for potential future program application.

I. INTRODUCTION

The timely delivery of aircrew flight simulators which accurately reflect the configuration of fleet aircraft has frequently been impeded by the late government delivery of aircraft data and aircraft/simulator common equipment to the trainer manufacturer. In an effort to alleviate this problem, the Navy has developed the acquisition strategy of "Buy Through The Prime" (BTTP). The purpose of this paper is to describe the process of buying through the prime, compare this process to conventional procurement methods, and to share lessons learned. It is important to note that the Navy execution of "buy through the prime" is still relatively immature and that the lessons learned represent more than two years of planning, but less than one year of execution. Future lessons learned, successes and failures will be candidates for subsequent papers.

II. DEVELOPMENT OF BTTP CONCEPT FOR THE A-6F/F-14D AIRCREW TRAINER SUITES (ATS)

A. Conventional Acquisition Process:

Before the development of the BTTP concept can be addressed, it is necessary to discuss, briefly, the conventional Navy process for acquiring simulators. For either approach, the first step is to establish a requirement. This is usually accomplished through the conduct of a front-end analysis or an Instructional System Development (ISD). The analysis of the skills to be taught, the student through-put requirements and the number of and location of training sites, etc., provides an initial estimate of the types and numbers of training devices/systems required. This requirement is refined and, finally, defined as the aircraft program proceeds through Full Scale Development (FSD). Once the training requirement is established, the Naval Air Systems Command (NAVAIR) is tasked by the Chief of Naval Operations (CNO) to procure and deliver the required training systems. NAVAIR, with the assistance and support of field activities and the cognizant Fleet Introduction Team (FIT) prepares specifications which define the performance requirements of the required training devices. Once the specifications are finalized and approved, NAVAIR proceeds through a routine synopsis/RFP/proposal preparation/source selection/contract award process. Normal program lead times for this conventional process vary from 24 to 30 months. Once the contract is awarded, periods of performance until Ready For Training (RFT) vary from 24 to 48 months, depending on the complexity of the training system (eg., Part Task Trainer vs. Weapon System Trainer). Therefore, the total time from establishment of a requirement to RFT can vary from 48 to 78 months.

Two of the deficiencies with the conventional process stem from the Navy's difficulty in procuring and delivering aircraft data and aircraft/simulator common equipment to the trainer manufacturer in a timely fashion. Because of the long, involved process of providing government approved aircraft data, the design of training systems is frequently based on preliminary aircraft design data resulting in the delivery of training systems which do not reflect the configuration of current Fleet aircraft. Additionally, the government frequently experiences problems in obtaining aircraft/trainer common equipment from the vendor who is providing that same equipment to the aircraft manufacturer (usually because of low quantities and low priority).

Not only do these problems cause late, out-of-configuration training systems, but they often generate corresponding claims for equitable adjustment from trainer manufacturers because the manufacturers are constrained in meeting specifications and/or delivery schedules.

B. Unique Aspect of the A-6F/F-14D Procurement

When the Secretary of the Navy directed the F-14D and A-6F aircraft development programs, NAVAIR was faced with the unique opportunity of procuring training systems for

two different Navy aircraft that would be using a significant amount of common equipment. Analysis revealed that the most efficient way to procure the suites of training systems, ensure the maximization of common equipment (to lower developmental and life cycle costs) and deliver the training systems to a realistic schedule and in a configuration representative of fleet aircraft, was to conduct a single procurement through the aircraft prime manufacturer - one contract, one prime.

C. Development of the Acquisition Strategy

Buy Through the Prime is based on four major tenets:

1. The aircraft prime manufacturer has immediate access to aircraft data. As aircraft designs are formalized, the prime can provide that data to the trainer manufacturer on a real-time basis. The trainer manufacturer does not have to await the laborious process of government approval of data before he can begin to design the training systems. The real-time provision of aircraft design data allows the trainer manufacturer to design, fabricate and deliver a training system to a substantially more mature baseline.

2. Because the aircraft prime is procuring equipment for use in the aircraft, it is a perfect conduit through which to procure aircraft common equipment for the trainers. For example, instead of ordering 24 "black boxes" for the aircraft being produced, the aircraft prime orders 30 to cover both aircraft and trainer requirements. The prime will then have the ability to adjust the priority of deliveries from the vendor and will also be able to take advantage of the economies of scale of the larger order. The savings in time and dollars can then be passed on to the government.

3. Because of the prime's involvement with the aircraft development and production programs, it is also the best source of configuration control for the training systems. The prime is in the driver's seat to establish a configuration baseline for the trainer manufacturer so that it may design and build fleet representative training systems.

4. Since the aircraft prime is responsible for "in-scope" schedule slips in the aircraft development program, it should also assume a corresponding responsibility for the training system, including associated cost impacts.

III. PROGRAM REQUIREMENTS

While the BTTP concept was being developed by the Navy, a parallel effort to define the training system requirements was also underway. An ISD was conducted by the prime and the results were reviewed and refined by NAVAIR, the Naval Training System Center (NTSC) and the cognizant FIT's. The final requirement called for two F-14D and three A-6F training sites with a total of seventeen training devices. Each F-14D site will be comprised of three Mission Flight Trainers (MFT's), a dual-dome Weapon System Trainer (WST) and one Tactical Environment System (TES) to link all five cockpits together for a single battle problem. Two of the A-6F sites are comprised of three WST's while the third site, for the USMC will have only one WST. Figure 1 reflects the devices, sites and associated acquisition milestones for the program.

IV. EXECUTION OF BTTP CONCEPT (CASE STUDY)

A. Training Systems Joint Working Group

In the March/April 1985 timeframe the Navy and Grumman established a Training System Joint Working Group (TSJWG) to ensure that both the A-6F and F-14D Phase I, Phase II, and Phase III efforts for Aircrew Trainers would be implemented with compatible goals, requirements, schedules, and deliverables. The efforts associated with the planned phases are defined as follows:

- Phase I - Instructional System Development (ISD) resulting in the requirements for the aircrew trainer suite.

- Phase II - Project Development resulting in the specifications and procurement package that defines the aircrew trainer suite.

- Phase III - The competitive solicitation and implementation of the deliverable aircrew trainer suite.

The TSJWG was co-chaired by NAVAIR APC205 and Grumman, and included full time representation from CNO and NTSC. The TSJWG met for the first time in May 1985, developed an approved Program Master Plan (PMP) in November 1985, and had its eighth and final working session in June 1986. During this period, guidance was provided with respect to the following items:

- Development of Suite/Device Specification concept

- Contract Data Requirements List (CDRL)

- Cost Reduction initiatives

- Aerodynamic Data Base commonality

- Phase III Budget Planning

- Warranty requirements

- Drawing requirements

- DOD-STD-2167 tailoring

- Reliability & Maintainability requirements

- ILS/CLS requirements

B. Program Master Plan

The PMP provided the necessary coordination between the separately funded A-6F and F-14D Phase I and Phase II efforts, with an appropriate focus on commonality and concurrency, as well as the formulation of the Phase III acquisition strategy that combined the implementation of the A-6F and F-14D aircrew trainers in a single contract. Phase III was subdivided into Phase A for the solicitation and source selection and Phase B for Development and Production of the aircrew

trainer suite.

A detailed Master Schedule was developed as an integral part of the PMP to provide the correlation between the Weapon System and the phased aircrew trainer milestones. This schedule is summarized and included as Figure 2, with planning dates as of November 1985. The actual dates for selected milestones are provided below for comparative purposes:

MILESTONE	PLANNED DATE	ACTUAL DATE
F-14D Phase I Complete	Jun 86	Sep 86
F-14D Phase II Complete	July 86	Sep 86
Phase IIIA Proposals Comp	Oct 86	Jan 86
Phase IIB Authorization	Jan 87	Apr 87

Even though the original target milestones were not achieved, partially as a result of incorporating a major ECP, the execution of Phase IIIB was completed within 8 months of the identification of Phase I Program requirements (a savings of 16 months over the conventional approach).

C. Unique Program Considerations

Rather than explore the total program from a case study perspective, this paper attempts to provide some discussion on those items that were unique. Some of the items represent techniques that are recommended for future programs, while others are discussed primarily to provide an awareness of a unique consideration.

1. In an attempt to make the planned short proposal preparation period more achievable for industry, the following additional formal events were included in the competitive solicitation process:

- The timely availability of aircraft data is critical to the preparation of detailed technical proposals for sophisticated aircrew trainers. In the case of emerging Weapon Systems like the A-6F and F-14D this type of data is not normally available to the trainer industry until it has been delivered and approved by the Government. As a result of the BTTP concept, Grumman was able to provide advance technical data on an incremental basis well in advance of the formal procurement package.

- Approximately six months prior to release of the formal procurement package a Request For Information (RFI) Conference was held to provide both a Weapon System and an Aircrew Trainer Suite overview, as well as respond to industry questions and inputs based upon the previously delivered RFI Data Package. This data package included aircraft data, preliminary Training Device specifications, and program acquisition planning information. The conference was held at NAS Oceana to allow all potential bidders to review the existing aircrew trainer suites for both the A-6E and F-14A programs.

- Approximately one month prior to the release of the formal procurement package a Pre-Invitation-To-Quote (ITQ) Bidders Conference was held to provide clarification of the final procurement requirements. The option pricing requirements would be an example of the information that was provided in detail at the Pre-ITQ Conference. Some of the option pricing considerations were:

  - FY88/89 Device production option exercise windows were defined separately as "expected" and "delayed" to allow maximum flexibility in case of funding delays. Refer to Fig. 1 for definition of alternate authorization windows.

  - In addition to combined pricing for the total program, stand-alone option pricing for A-6F and F-14D options were required to provide for the possibility of a major aircraft program delay/cancellation.

Without these formal advance coordination actions, it is highly unlikely that industry could have responded in such a short time for a program of such complexity.

2. Another unique requirement that was provided during the RFI timeframe was for industry to form teams in support of a co-developer concept that required each team member to:

- split the work approximately 50-50
- freely and openly share all data
- be prepared to compete against each other for future modification orders.

This approach was evaluated during the RFI question and answer process, and was generally accepted by industry prior to becoming a requirement of the ITQ.

3. In the area of trainer support, this program required that the trainer manufacturer provide all Maintenance Material and personnel as part of its firm fixed price in consonance with the Navy's policy of total contractor logistics support.

4. In addressing the problem of delivering a training device concurrently and in the same configuration as the first production aircraft, the Navy and Grumman created a planned configuration update. This update is designed to match the configuration of the trainer to the delivered production aircraft. The effort associated with the update development and field incorporation is to be included in the fixed price for the Trainer Full Scale Development (FSD).

5. In support of defining an executable program, Best And Final Offer (BAFO) pricing guidelines identified items for deferred option pricing so that a program could be constructed that was compatible with existing budgets yet have provisions for adding deferred items at a fixed price when funding became available. After receipt of the BAFO, due to a funding shortfall, it became necessary to defer certain support items for funding in conjunction with

the production options. Since this was anticipated, the reduced program was executed without an additional BAFO iteration and associated schedule delay.

V. LESSONS LEARNED

One of the primary purposes of this paper is to share "Lessons Learned" during the various stages of the program to date, so that future programs may evaluate them for potential application in their advance planning. Some of the key considerations and recommendations are summarized below:

A. Correlation of ISD and Specification Development

As clearly depicted in Figure 2, the Phase I ISD and Phase II Specification efforts were planned and scheduled as essentially parallel efforts. This schedule was driven by the requirement to have specifications available in mid-1986 to field the initial training devices in late 1989.

This scheduling was tolerable on this program only because Grumman was responsible for both the ISD and specification efforts, and was able to informally share ISD data in support of the specification development. Even so, it is clearly not the most effective way to plan the program. It would appear in retrospect that given the primary requirement for specifications in mid-1986, it would have been more appropriate to tailor the ISD effort to the time available, so that fleet approved functional requirements could be provided no later than the end of 1985. This would have allowed the specification development to proceed with a firm approved baseline.

B. Correlation of Contractor Developed Procurement Package with Government Contract

In addition to the specifications developed during Phase II, Grumman was also required to develop the total procurement package for the Phase IIIA competitive solicitation as part of the Phase II effort. Once again, as indicated in Figure 2, the schedule did not allow adequate time for the Government to utilize the delivered procurement package as the basis for their internal procurement requisition for the Phase III Contract. As a result, the Grumman procurement package and the government procurement requisition were developed in parallel, and were made compatible only after an extensive series of in-process coordination meetings. Since the procurement package is essentially independent of the specification effort, its preliminary delivery should be scheduled so that it supports the preparation of the Government Purchase Requisition.

C. Consideration of Dual Developer through PDR/CDR

During the development of the Equipment Procurement Plan in the Phase II timeframe, the potential of extending the competitive solicitation process for the two leading bidders into the design phase was given some consideration. The concept was discarded primarily because it was introduced too late in the program and there were potential cost and schedule penalties associated with that approach. Future Programs that are being planned in support of emerging weapon systems should consider the alternate approach of dual developers at the initial program planning stage, and make the decision based upon analyses of Weapon System maturity, Budget availability, schedule impact, etc.

D. Proposal Evaluation Scoring Considerations

The basis for award was best value to the Government/Grumman, with the proviso that Cost shall be at least equal in value to the sum of the Technical, Management, and ILS elements. Even though it did not occur in this program, the potential did exist for a bidder to "buy in" without any penalty for cost realism. For future "Best Value" programs, it would appear reasonable for the Cost Volume to be evaluated and scored similar to the technical volumes, so that cost realism could be factored into the overall score.

The other area for scoring consideration results from the technical, management, and ILS scores improving as a result of the ongoing Deficiency Report (DR) and Clarification Report (CR) process. It is felt that future proposal evaluation plans should provide some weight to the initial score of the various areas, in addition to the final score that results from the DR/CR process.

E. Coordination of Navy/Grumman Effort

The potential for duplication of effort in any BTTP program is significant. In areas such as specification development and proposal evaluation, the government generally utilizes its own resources. When these efforts are assigned to a Contractor such as Grumman, there should be a clear definition of the Government role as part of the basic contract. This is particularly critical with respect to providing the government with adequate visibility into the proposal evaluation process and also maintaining the necessary security required by the contractor's procurement department. The degree of government monitoring and control should vary from program to program based upon unique program requirements, but the degree of involvement should be clearly stated as part of the contract to eliminate any confusion and avoid any unnecessary duplication of effort. The goal should be for the government to reduce their effort in a BTTP program by a minimum of 50% over that expended on a traditional acquisition.

VI. COST CONSIDERATIONS

Once the benefits of BTTP are discussed and understood, the natural question is: "How much will it cost?". In the case of the A-6F/F-14D ATS procurement, the total additional cost for prime participation (direct labor, G&A, overhead, profit and sub-contract burden) amounts to less than 20% of the entire training system procurement. For that amount, the Navy will be assured of the on-time delivery of trainers which accurately reflect the configuration of current fleet aircraft. The prime has assumed the risk for the timely delivery of aircraft data and aircraft/trainer common equipment. He has also assumed the

responsibility for delivering trainers in current fleet configuration, as well as the responsibility for in-scope schedule slips. Additionally, the BTTP approach relieves subcontractors of the cost of an associate contract with the prime (for data) and permits a lower bid. The cost of BTTP is easily offset by the effort and risk assumption of the prime.

VII. SUMMARY

As indicated in the introduction, this case study reflects the results of a BTTP program primarily during the planning and acquisition phases. To this extent, the program goals of commonality and concurrency are being achieved, but the true test will be during the implementation phase. The evaluation and lessons learned during the design phase would be an appropriate extension of this case study for future consideration.

ABOUT THE AUTHORS

Commander Jarrott is a Naval Flight Officer and holds a Bachelor's Degree in Business Management and a Master's Degree in Safety. He has completed the Defense Systems Management College Program Managers Course and is a designated Weapon System Acquisition Manager (WSAM). Commander Jarrott has accrued four and one-half years of acquisition experience, most recently, two and one-half years in training system acquisition at the Naval Air Systems Command.

Mr. MacLeod holds a Bachelor's Degree in Mechanical Engineering and has over twenty years of Engineering and Program Management experience with Grumman. The last fifteen years have been directly associated with the Acquisition Management of Aircrew Trainers, with the recent assignment as the Program Manager responsible for the A-6F/F-14D Aircrew Trainer Acquisition. Mr. MacLeod is currently the Resident Program Management Technical Representative for the ATS Program at AAI Orlando, Florida, as well as the NTSC Program Interface in Orlando for all A-6 and F-14 Grumman Trainer Activity.

F-14D/A-6F AIRCREW TRAINER ACQUISITION SCHEDULE — May 1987

PROGRAM	TRAINER	E/D*	SITE	PDR	CDR	I/O*	AUTHORIZATION	READY FOR TRNG	DURATION
F-14D	Mission Flt Trnr #1		Miramar	9Mos	18Mos	I	5/87	8/90	40M
F-14D	MFT #1 Config. Updte		Miramar			I	5/87	**	
F-14D	Mission Flt Trnr #2	(E)	Miramar			O	11/88 - 2/89	11/91	36/33M
F-14D	Mission Flt Trnr #2	(D)	Miramar			O	3/89 - 2/90	11/91 - 10/92	32M
F-14D	Mission Flt Trnr #3	(E)	Miramar			O	11/88 - 2/89	1/92	38/35M
F-14D	Mission Flt Trnr #3	(D)	Miramar			O	3/89 - 2/90	1/92 - 12/92	34M
F-14D	Mission Flt Trnr #4		Oceana			O	11/91 - 2/92	5/94 - 8/94	30M
F-14D	Mission Flt Trnr #5		Oceana			O	11/92 - 2/93	5/95 - 8/95	30M
F-14D	Mission Flt Trnr #6		Oceana			O	11/92 - 2/93	7/95 - 10/95	32M
F-14D	Weap Sys Trnr DD #1		Miramar	9Mos	18Mos	I	5/87	4/91	48M
F-14D	WST #1 Config. Updte		Miramar			I	5/87	**	
F-14D	Weap Sys Trnr DD #2		Oceana			O	11/91 - 2/92	11/94 - 2/95	36M
F-14D	Tact Envirn Sys #1		Miramar	9Mos	18Mos	I	5/87	8/91	52M
F-14D	Tact Envirn Sys #2		Oceana			O	11/92 - 2/93	11/94 - 2/95	24M
A-6F	Weap Sys Trnr #1		Whidbey	10Mos	18Mos	I	5/87	8/90	40M
A-6F	WST #1 Config. Updte		Whidbey			I	5/87	**	
A-6F	Weapon Sys Trnr #2	(E)	Oceana			O	5/88 - 8/88	2/91 - 5/91	33M
A-6F	Weapon Sys Trnr #2	(D)	Oceana			O	9/88 - 2/89	6/91 - 11/91	33M
A-6F	WST #2 Config. Updte		Oceana			O	5/88 - 2/89	**	
A-6F	Weapon Sys Trnr #3	(E)	Whidbey			O	11/88 - 2/89	8/91 - 11/91	33M
A-6F	Weapon Sys Trnr #3	(D)	Whidbey			O	3/89 - 2/90	12/91 - 11/92	33M
A-6F	Weapon Sys Trnr #4		El Toro			O	11/89 - 2/90	8/92 - 11/92	33M
A-6F	Weapon Sys Trnr #5		Oceana			O	7/91 - 2/92	4/94 - 11/94	33M
A-6F	Weapon Sys Trnr #6		Whidbey			O	11/91 - 2/92	8/94 - 12/94	34M
A-6F	Weapon Sys Trnr #7		Oceana			O	11/92 - 2/93	8/95 - 11/95	33M

Notes: * I - Initial Authorization E - Expected
O - Option Follow-On D - Delayed
** No later than 12 months after device RFT or first production aircraft delivery, whichever comes later

A-6F/F-14D ATS PROGRAM MASTER SCHEDULE - SUMMARY 11/85

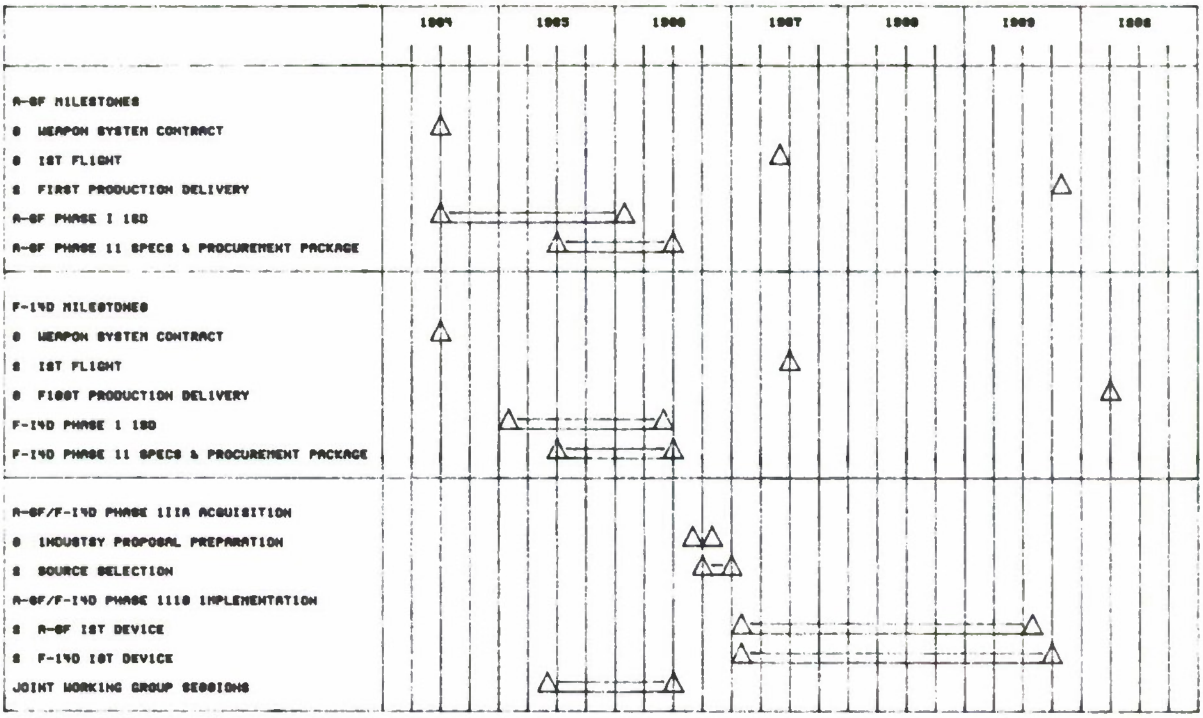


FIGURE 8

COMMERCIAL ACQUISITION OF AN AIR COMBAT SIMULATOR

Dorothy M. Baldwin
General Dynamics/Fort Worth Division
P. O. Box 748, Mail Zone 5944
Fort Worth, Texas 76101

ABSTRACT

Considerable emphasis is being placed on resolving cost and schedule problems associated with military procurements. Commercial acquisition of military type equipment can often result in significant cost and schedule savings, with little or no compromise in performance. This paper describes strategies which can be used to reduce risk, schedule and cost factors during a commercial acquisition. Some of the techniques which will be discussed include: early communications on details of the requirements including extensive technical reviews prior to contract award; definition of acceptance test criteria prior to contract award, clear definition of the buyer/seller interface including all facility requirements; procurement of major subsystems separately; use of commercial quality rather than military standards, deletion of in-plant acceptance test, and utilization of proven systems.

The specific procurement of a dual-dome air combat simulator environment will be used as an illustration of the techniques discussed. The acceptance testing for the first dome was completed thirteen months after contract award. The acceptance testing for the full dual dome system was completed within twenty months after contract award or two months ahead of schedule. No cost overruns occurred during this procurement. Typical schedules for delivery of government procured Weapon Systems Trainers have been 36 to 48 months. In the case described here, the simulator buyer was also the airframe manufacturer and therefore had strong economic and technical interest in the simulator procurement. Also, the ownship cockpit simulation was done by the buyer. The paper is written from the buyer's perspective.

While all strategies employed in this commercial acquisition may not be applicable to government training system acquisitions, the author believes that a review of some of the pertinent details of this procurement may provide concepts of interest to simulation manufacturers, as well as to the government procuring agencies.

INTRODUCTION

General Dynamics Flight Simulation Laboratory has requirements to provide aircraft designers with the opportunity to have experienced pilots "fly" and evaluate proposed aircraft designs or concepts, in simulations that are complete with fully operable avionic controls, displays, and realistic operationals mission scenarios. These evaluations are conducted prior to detailed production design commitments. The simulators are used to collect pertinent data related to aircraft performance, handling qualities, controls, weapon delivery, etc. This data is then analyzed for input to the aircraft design process.

Although simulation has long been an integral part of the aircraft design process, the required technology to provide a real-time environment for pilot-in-the-loop evaluation of realistic tactical mission scenarios is still advancing at a rapid pace. The Fort Worth Division of General Dynamics has developed extensive real-time flight simulation capabilities in their new flight simulation laboratory. In particular, a dual-dome Air Combat Simulator (ACS) has recently been added to the simulator facility. (see Figure 1.)

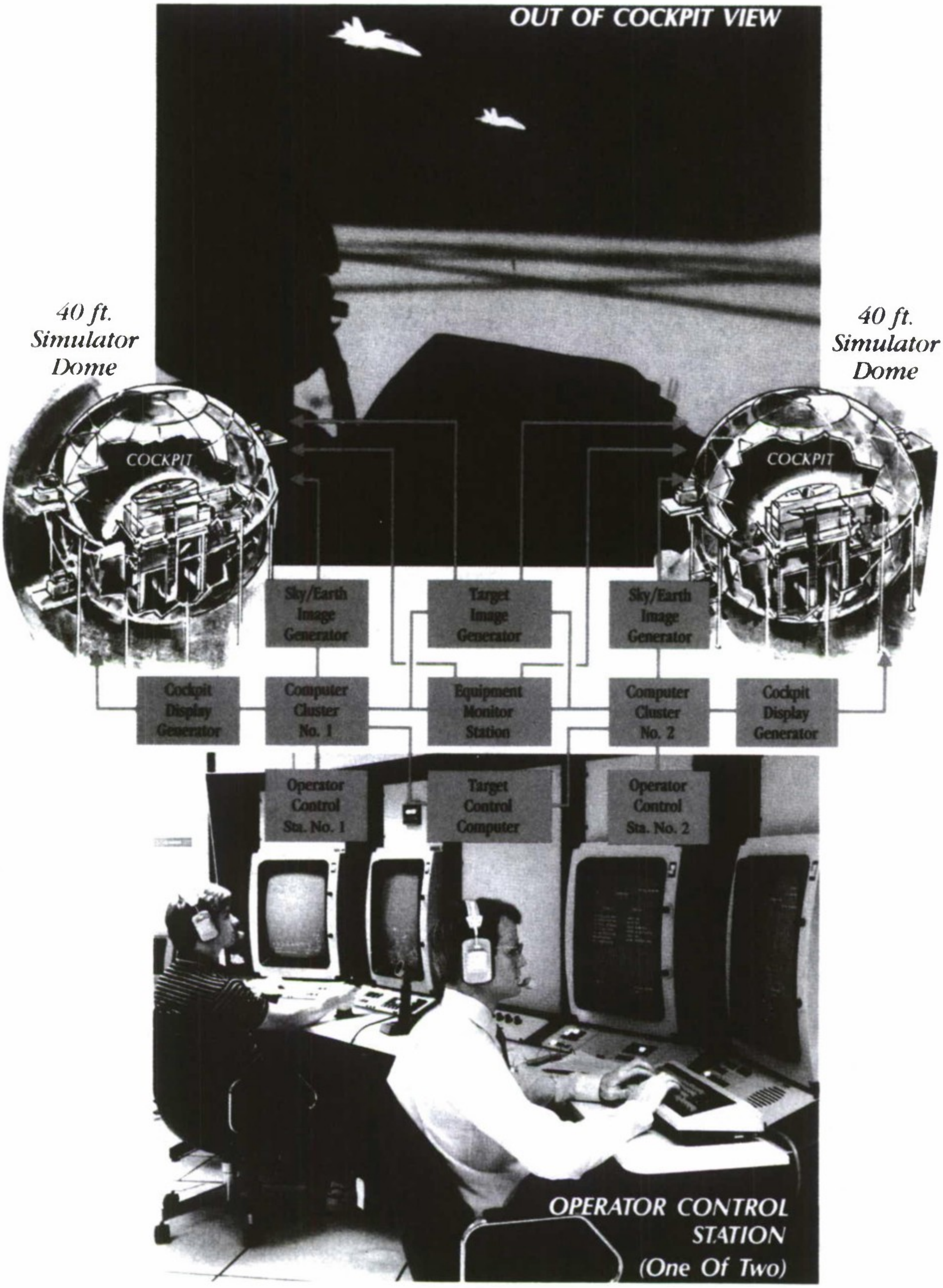


FIGURE 1: DUAL DOME SIMULATOR

The latest developments in simulation technology allow fighter aircraft system designs to be evaluated in the arena where it counts most - AERIAL COMBAT!

The program need schedule for an air-to-air combat simulation capability was extremely short. The decision was made to concentrate General Dynamics simulation efforts in the area consistent with the primary business of the Fort Worth Division, i.e., the design of aircraft. GD designers would concentrate on the simulation of the aircraft itself, including flight dynamics, aerodynamics, avionics, and all the hardware associated with the cockpit, including interfaces. A conscious decision was made not to enter the environment simulation fields, i.e., not to develop in-house capability in building visual displays, image generators, computers, domes, radar simulator, etc. Essentially, General Dynamics was following the reasoning of Isaac Newton who said in the 17th century, "If I have seen farther than others, it has been by standing on the shoulders of giants." There were enough "giants" in the environmental simulation field already.

The basic requirements for the simulator capability were outlined and capabilities discussions were conducted with potential contractors. The critical requirements of the air-to-air simulator system were as follows:

FUNCTIONAL REQUIREMENTS

- o Dual Dome Simulation Stations
- o High Resolution (Eye Limited) Targets - Two Per Dome, Full 360-degree Field of Regard
- o Generic Sky/Earth Scene; Full 360-degree Field of View, CIG Imagery
- o Host Computer System Capable of Multi-Airframes, Missiles, Target Geometry Calculations

PROGRAM REQUIREMENTS

- o Off-the-Shelf Design for Minimum Risk & Schedule

The basic strategy used was to locate an existing system, as close to the General Dynamics requirements as possible, and then to make modifications, only as necessary, to meet the unique General Dynamics requirements. GD test pilots, who would be the ultimate system users, flew and evaluated candidate systems and provided input to technology trade-off considerations.

The Air Combat Simulator procured by General Dynamics is a derivative of the 2E7 built by Hughes Aircraft Company as an F/A-18 trainer.

Requirements and Final Performance

No compromises were made in system performance requirements. As examples of the type of requirements imposed and met by the delivered system: Table I summarizes the specified performance of the visual sky/earth subsystem; Table II summarizes the performance of the visual target subsystem; Table III summarizes the required modes of the operator control station.

Contractor/Subcontractor Relationship

The dual dome Air Combat Simulator is a system that features two independent pilot stations, capable of operation in either a stand-alone mode or in an integrated mode. Figure 1 has a block diagram showing the major ACS components. The principal components of the ACS are two 40-foot diameter domes with associated optics, image generation and graphics equipment, pilot stations located one in the center of each dome, two operator control stations, computer complex, equipment monitor station, and facility support equipment (power distribution, air conditioning, etc.). Major components of the ACS and subsystem integration were provided by the Hughes Aircraft Company. Final integrationa of the ACS (with a cockpit) was done by General Dynamics.

The GD procurement approach was to minimize technical and delivery schedule risk. Therefore, the subsystems used represented existing or slightly modified designs for "off-the-shelf" equipment from existing simulation devices. Provisions were made to reduce acquisition cost by allowing GD to purchase and provide as GDFE (General Dynamics Furnished Equipment), large off-the-shelf non-contractor produced items, i.e., the Computers, Target Image Generator, Domes. The computer configuration was agreed to by GD and Hughes engineers prior to contract award, and GD then procured the computers directly from Gould. The target image generator, a CT-5 was procured by GD directly from Evans & Sutherland. Two 40 foot diameter domes were procured by GD directly from Spitz.

In all three cases, the specifications were agreed to by GD and Hughes prior to contract award to Hughes. Hughes acted as integrator for the components provided by Hughes, Gould, E&S, and Spitz. GD monitored all four contracts.

GD had complete responsibility for the cockpit. Because of the company proprietary nature of the program involved, no information concerning the aircraft, which would utilize the 40' domes, was transmitted to Hughes. This proved to be a challenging matter for GD engineers. One approach used to reduce the difficulty of this approach was to require that GD software and hardware engineers be given long-term on-the-job training at Hughes during the contract performance period. These GD engineers were then tasked with transferring their newly gained knowledge of the Hughes simulator to GD engineers responsible for integratinag the GD aircraft into the simulator. This strategy proved successful. In fact, when the first dome passed acceptance testing 13 months after signing the contract, GD engineers had a cockpit waiting in the hallway to "roll in". The cockpit was up and flying within one month!

Changes to the Existing System

As was previously stated, changes to the existing system (2E7) were kept to an absolute minimum to reduce schedule and cost impacts. However, one area requiring change was the computer configuration. The 2E7 utilized 13 Gould 32/67 series computers (six per dome with one for interface). Each computational cluster has a combined throughput of six (6) MIPS. This was insufficient for an engineering type simulator. For the GD configuration, three (3) Sel 32-9780 processors networked through a shared memory module and one FPS 5000 are utilized for each dome. Each Gould complex provides a total throughput of 27 MIPS for each dome cluster. The two computer clusters can be interfaced for integrated operation, i.e., both domes fly together in air-to-air combat.

Based on GD test pilot evaluation of the existing Hughes simulator, GD opted not to utilize the G-seat and not to go to the expense of sinking pilings to stabilize the light valve projection towers. Vibration tests done at GD-FSL, prior to contract award, also indicated that vibration stabilization was not necessary.

In order to do integrated acceptance testing, without a cockpit in the dome, a unique approach had to be developed. A "cockpit simulator" was developed to exercise system software and hardware. The cockpit simulator is engaged from each Operation Control Station (OCS) and uses OCS controls/displays to simulate a simplified generic air-frame. The cockpit simulator permitted checkout and test of the visual system, aural system, environmental simulation, threats, weapons, ACS modes, and interfaces. The cockpit simulator was also considered to have the potential of being a valuable design tool after acceptance of the simulator.

In addition, some minor changes were made, such as requiring P53 phosphor rather than P43 phosphor on the CRT target projectors to reduce CRT burn.

Contract Management Team

The contract management team was kept as small as feasible. A Program Manager/Technical Lead from the FSL provided the focal point for the GD team. After contract negotiations, a GD buyer was the only non-FSL (Flight Simulation Laboratory) team member. The GD Program Manager/Technical Lead coordinated the efforts of technical experts in the following areas: Software, computer hardware, visual simulation, operations and maintenance, program requirements. A key point here is that the same personnel: (1) provided input to the specification, (2) helped in monitoring the contract by attending reviews and reviewing preliminary documentation, (3) attended on-the-job training at the vendor's facility, (4) helped with acceptance testing, (5) and are the final end users. This is very important, because a "sense of ownership" was developed, which aids considerably. Also, changes in requirements or misunderstandings as to end usage are greatly reduced.

Interface Considerations

A key element which led to success in this procurement, was the time and effort undertaken, prior to contract award, to clearly define the overall division of responsibilities between GD and Hughes in implementing the ACS. Proposed responsibilities were described for hardware elements, software elements, data items, installation, integration and test and all other deliverables. Table IV provides an example for Facilities Division of Responsibility. All other areas of the procurement had division of responsibilities clearly defined. Since the dual-dome simulator was being installed into an existing building, the facility interface considerations were not minor (see Figures 2 and 3). The building was provided by GD. However, prior to contract award, Hughes was required to do a facilities survey and define modifications required to the GD facility. Major modifications were then documented and included in the contract as a "Preliminary Facilities Requirements Document". GD then produced a detailed design to implement the required facility modifications. A final facility modification design review was held one month after contract award.

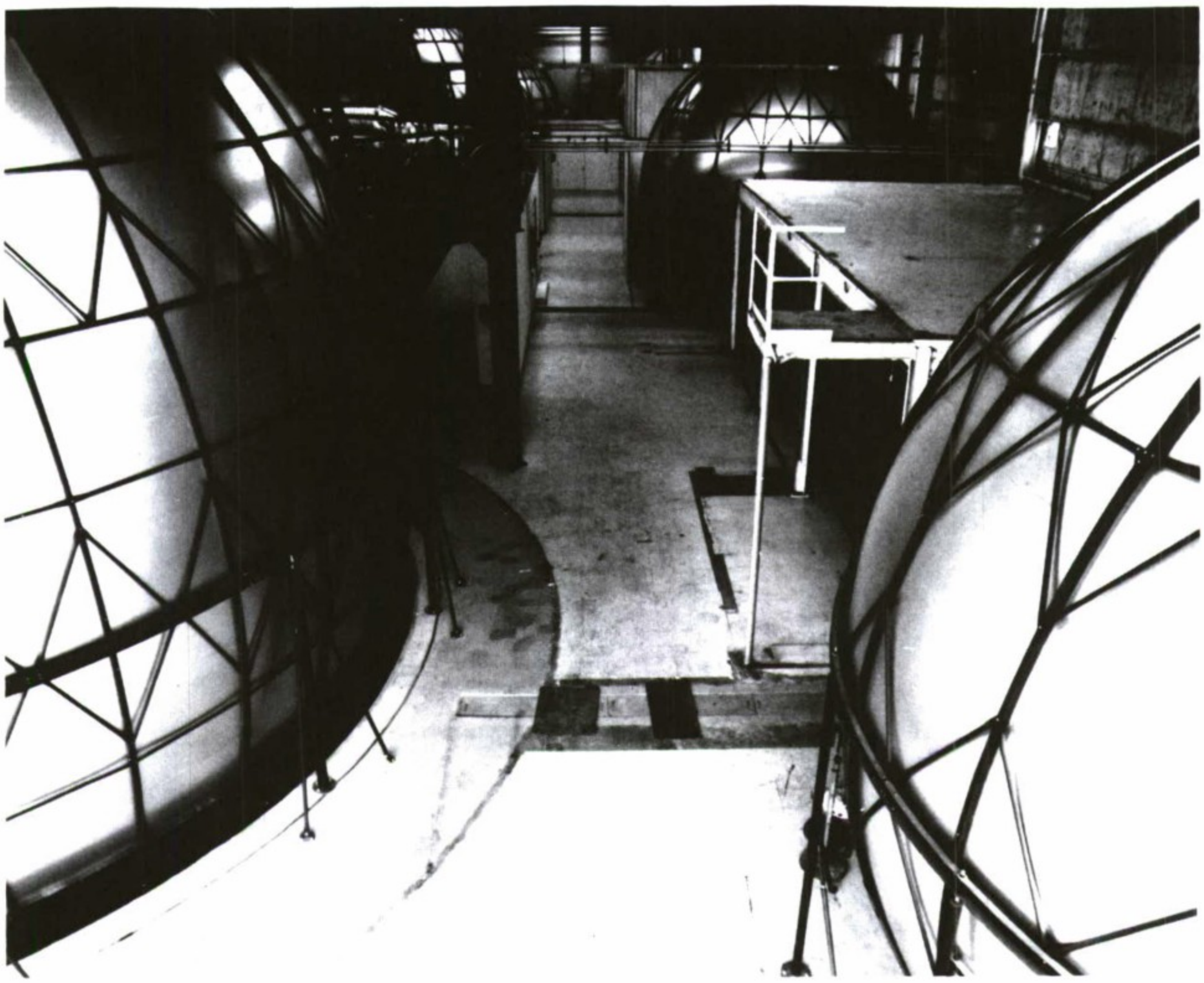


FIGURE 2

GD-FSL Facility under construction: showing two 40 ft-diameter domes used for Air Combat Simulator along with other domes being installed and integrated for other engineering simulation usage.

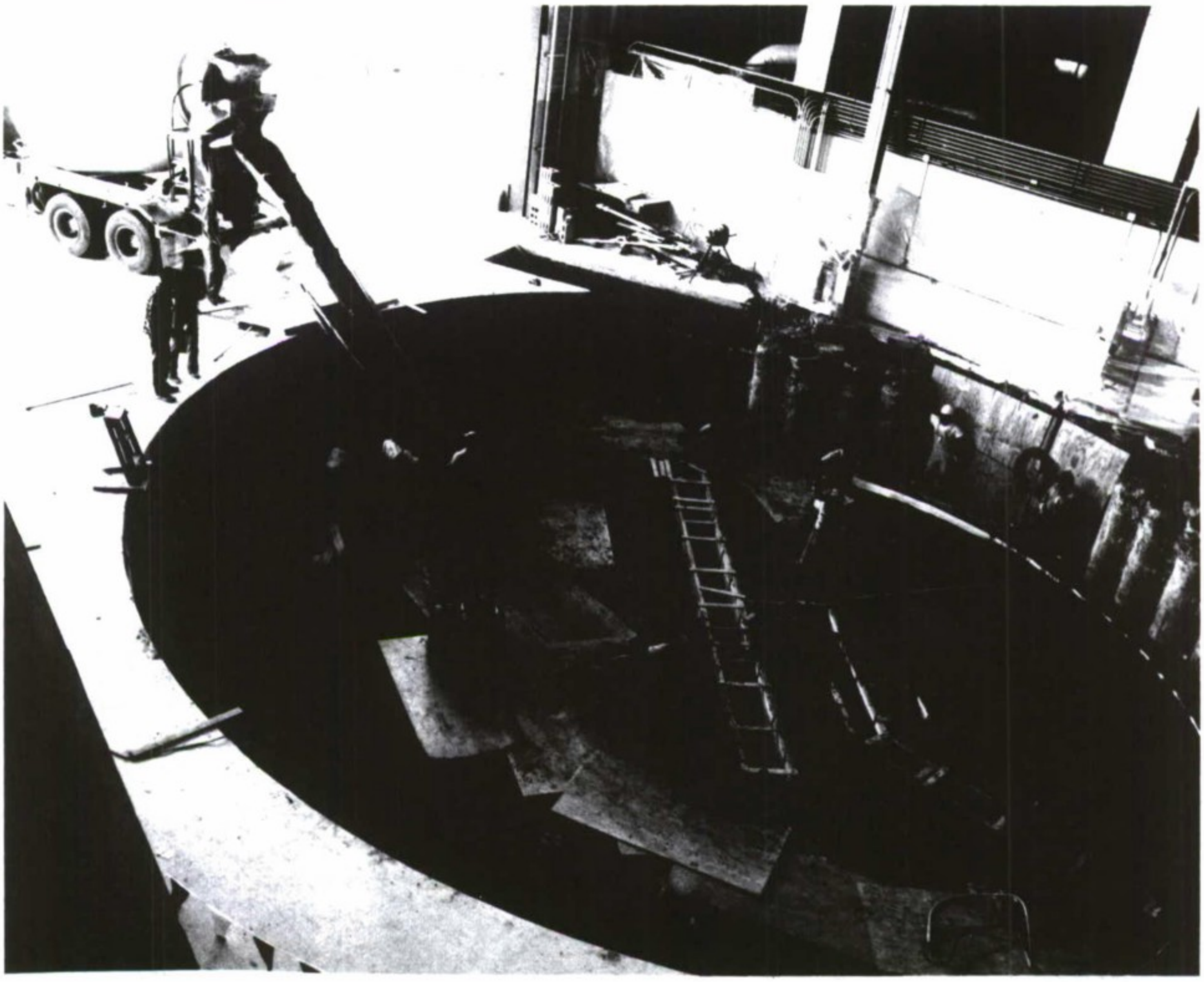


FIGURE 3

Preparation of one section of the GD-FSL facility prior to installation of one of the 40 ft. diameter domes.

Commercial Quality/Warranty vs. Detailed Manufacturing Surveillance

Quality assurance provisions in the contract were in accordance with best commercial practices. In addition, maintainability and reliability were handled through a warranty. GD did not devote extensive time or effort to managing the standard elements of producing a simulator (manufacturing procedures, configuration management, reliability/maintainability, logistics support, etc.). It is felt that this approach saved considerable schedule time and contract monitoring cost, while not proving detrimental to the quality of the delivered product. In fact, availability of Dome #1 has been approximately 97% since completion of acceptance testing.

Acceptance Testing

The terms of the acceptance testing were agreed to prior to contract award. This included agreement as to: (a) number of tests required, (b) subsystems to be test, (c) general scope of each test. This allowed for a more realistic cost and schedule to be provided. In addition, all acceptance tests were done on-site at the buyer's facility which eliminated the time and cost of integrated acceptance testing at the seller's facility.

Documentation

Technical data cost savings resulted from the following:

(a) Data was prepared according to best commercial practices.

(b) Data was accepted which was modified versions of existing data, as long as it accurately reflected the simulator subsystem delivered to GD.

In general, the emphasis was on receiving all data necessary to operate, maintain, and modify the simulator; without too much concern as to the exact format of the data. This is feasible when the data is reviewed and accepted by the end users.

CONCLUSIONS

How does this experience in procuring an engineering simulator relate to the training simulator procurement process? The author does not wish to appear presumptuous by reiterating things which perhaps are already widely known. The author also expresses concern lest the ideas expressed here be literally applied to government acquisitions. The commercial practices described here worked because of the commercial environment in which they were applied. Government contracting requires use of a different acquisition philosophy and the procedures described may not be applicable without changes in that philosophy. With these caveats in mind, a summary of schedule and cost saving strategies utilized in this procurement are outlined below:

(1) Have as clear a definition as feasible of the intended use of the simulator so that appropriate technology-cost-schedule tradeoffs can be made. This includes consideration of the physical location of the simulator and its operation and maintenance requirements.

(2) Do not re-invent the wheel. Use existing technology, when existing technology will do the job. This procurement has demonstrated that it is feasible to take an existing simulator, and on a very short schedule, substitute a completely different aircraft into the same simulator environment.

(3) Involve end users early in the procurement, i.e., when the specification and statement of work are being prepared; and keep end users involved throughout the procurement, acceptance and operation of the simulator.

(4) Keep the management and technical monitoring team small and maintain continuity by maintaining the same team through the procurement.

(5) Accept commercial quality on equipment and substitute warranties in place of detailed monitoring of manufacturing methods.

(6) Recognize that contractors are entitled to a reasonable profit, and work with them to reach a mutually beneficial agreement.

(7) Spend the time and energy to work out a highly detailed agreement prior to contract award, including a clear definition of what constitutes acceptable performance by the contractor.

TABLE I
SKY/EARTH SUBSYSTEM PERFORMANCE

DISPLAY

- 360 H x 140 V FOV
- 0.15 FT L
- 18 ARCMIN (RESOLUTION)
- NON-DISCERNIBLE OPENING

COMPUTER IMAGE GENERATOR

- 6 CHANNELS
- 3300 3D VECTORS
- MAPPING CORRECTION
- 3 COLOR
- 1023 LINE
- DATA BASE MANAGEMENT
- WEAPON EFFECTS
- ATMOSPHERIC EFFECTS

TABLE II
TARGET SUBSYSTEM PERFORMANCE

DISPLAY

- 2 VISUAL TARGETS/MODE
- 360 H x 140 V FOR
- 23 FOV - MAX
- 1 ARC MIN/LP - MAX
- 2.4 FTL - MAX
- PROJECTORS OUTSIDE FOV

CIG

- CT5
- 250 POLYGONS/TARGET
- MONOCHROME
- DISTORTION CORRECTED

TABLE III
OPERATOR CONTROL STATION MODES

MODE	CAPABILITY
FLIGHT	SINGLE PILOT NORMAL FLIGHT OPERATIONS, OPERATOR FLOWN OR AML TARGET
INTEGRATED	LINKS 2 DOMES
RECORD/ REPLAY	REPEATS (Visuals, A/C Motion, Controls/Displays, Audio)
DEMO	PLAYS A "CANNED" SCENARIO
PLAN	SET UP INITIAL CONDITIONS
OPERATOR TRAINING	PROGRAMMED INSTRUCTION FOR USE OF OCS, ACS
TEST	TEST ALL MAJOR ACS SYSTEMS IN SEQUENCE, OR INDIVIDUAL SUBSYSTEMS
COCKPIT SIMULATOR	OCS AS SIMULATED OWNSHIP,SUBSYSTEM INTEGRATION, DESIGN TOOL

TABLE IV - DIVISION OF RESPONSIBILITY

SELLER RESPONSIBILITY	BUYER RESPONSIBILITY
Facilities	**Facilities**
o Installation of System Power/Cable	o Equipment Cooling
o Submit Facilities Requirements	o Dome Access
o Provide hoists, cranes, and other installation aids required for the cockpit support platforms, light valve towers, target projectors, and sky/earth projectors.	o Pit Modifications for Access/Projectors
	o Raised Floors
	o Overhead Crane
	o Lighting
	o System Ground
	o Standard machine shop aids such as hand trucks, fork lifts, etc., be made available periodically by GD for transport and installation of other heavy equipment.

ABOUT THE AUTHOR

DOROTHY M. BALDWIN has been A SENIOR ENGINEERING Specialist in the Flight Simulation Laboratory at General Dynamics/Fort Worth Division for the past 3 1/2 years. During that time her assignments have included: Program Manager/Technical Lead for Procurement of a dual-dome Air Combat Simulator environment; Program Manager/Technical Lead on a Tank Simulator development; Lead Simulation consultant for a foreign customer; and technical consultant in visual and sensor simulation for the F.S.L. expansion. Prior to joining G.D., Ms. Baldwin worked as a physicist at the Naval Training Equipment Center where her assignments included: Principal Investigator on the Multi-Spectral Image Simulation Project; Principal Investigator on the Active Screens Project; Optics work on the 360 degree Laser Scan Projector, Helmet Mounted Display system and Annular Projection System. Ms. Baldwin has also been a physics instructor at Kent State University and Florida Technological University. She obtained her M.A. in Physics from Kent State University in 1968 and her B.A. in Physics from Hartwick College in 1965.

CONCURRENT TRAINER AND AIRCRAFT DEVELOPMENT

Michael A. Frankie
MCAIR Training Systems
McDonnell Douglas Corporation
1187 Corporate Lake Drive
Saint Louis, MO 63132

ABSTRACT

The current requirements to deliver Aircrew Training Devices concurrently with aircraft weapon systems development has placed major issues in front of trainer manufacturers. There are various documents (MIL-Specs, DIDs, Specifications, etc.) that describe the approach as to how training devices will be procured, however, implementation of these approaches is not as simple as it has been. Previously, trainers were considered after the weapon system had been developed permitting several issues to have been resolved. This paper will look at these issues from the contractor's point of view and present some areas that the customer, contractor and weapon system prime need to be fully cognizant of during concurrent trainer and aircraft development.

INTRODUCTION

Current and future trainer procurement schedules are requiring aircrew trainer systems to be developed and delivered concurrently or ahead of new aircraft development. Not only is the customer expecting a timely delivery of the training device but also a device that is in configuration with the production aircraft that are being delivered. To accomplish this task, the specifications are being written such that the Aircrew Trainer Device (ATD) is based on a given design basis aircraft corresponding to a given tail number from the production lot (Ref. 1). These requirements present a philosophical question to the trainer manufacturer. Does he design and build a trainer based on a design freeze utilizing the best data available at that time and then propose an update to the production configuration after the trainer is delivered; or does he design and build a trainer with a pseudo design freeze and attempt to incorporate as much as possible before delivery in order to match the aircraft configuration?

The customer undoubtedly desires to have the trainer delivered in configuration and to not have to accommodate an immediate update of the ATD. In order to accomplish the delivery of the ATD on schedule and in configuration, the design criteria for the production aircraft is required. In reference 2, Lt. Smith outlines the approach needed to develop the criteria list for a trainer device and states 'a simulator contractor should have an essentially complete DCL (Data Criteria List) not later than completion of preliminary design.' While in theory, his concepts are totally valid, in practice it is usually not possible to fully develop the DCL. Most of the criteria required in the DCL is not published early enough for the trainer contractor to include or even be aware of its existence.

The information presented in this paper is based on McDonnell Aircraft's experience in developing and delivering the AV-8B aircrew trainers on schedule and in configuration with the design basis aircraft. Although McDonnell Aircraft was the prime contractor for both the aircraft and the trainer, the trainer development was under a separate contract and the point of view presented herein is that of a trainer contractor, not that of the aircraft prime.

ISSUES

In the past, aircrew trainer procurement was usually considered after the development of the aircraft was well under way or completed. This time factor allowed for three significant items to occur; 1) the flight dynamics detailing the aircraft flying qualities and performance had been tested and documented, 2) the aircraft systems design had been refined and was fairly stable and 3) the fleet users had some experience in the aircraft and knew the subtleties of the aircraft that would be required in a training device. With the desire to have the trainers concurrent with the delivery of the first production aircraft, none of these three items have occurred for new airframes.

FLIGHT DYNAMICS

Flight testing of an airframe is a time consuming task. A building block approach is utilized to gradually expand the envelope of testing without taking undue risks. As problems are encountered, design changes are made to overcome the problems (i.e. re-rigging of control surfaces to minimize drag in a given flight regime). The upstream effect due to the change as well as the downstream effect must be verified. Thus, some of the previously known characteristics may have changed due to a given modification. Additionally, a certain amount of time is required to analyze and document the results from flight testing the airframe. Although a particular flight regime may be tested in a given time period, it may be six to twelve months before the data is actually published in a usable form (see Figure 1).

The format in which the flight test results are documented is sometimes insufficient or inappropriate to utilize as criteria data for verification of the trainer device. Trainer specifications typically detail the format of the criteria by way of the tests to be performed. Additional data is required that may not exist at the time it is needed for the trainer or may never exist. A flight test program and the requirements placed on it are to verify the characteristics of the airframe. What may be qualitatively tested on the aircraft may be required to be quantified in the trainer. Examples of this are nosewheel steering characteristics and taxi speeds of the aircraft. These areas are seldom documented to the extent required to quantify a trainer. In the past, there existed enough knowledge among the users to develop a consensus on which to base the trainer. With a limited number of contractor and customer test pilots, it is difficult to generate a consensus to develop the criteria for trainer applications in the areas of qualitative tests.

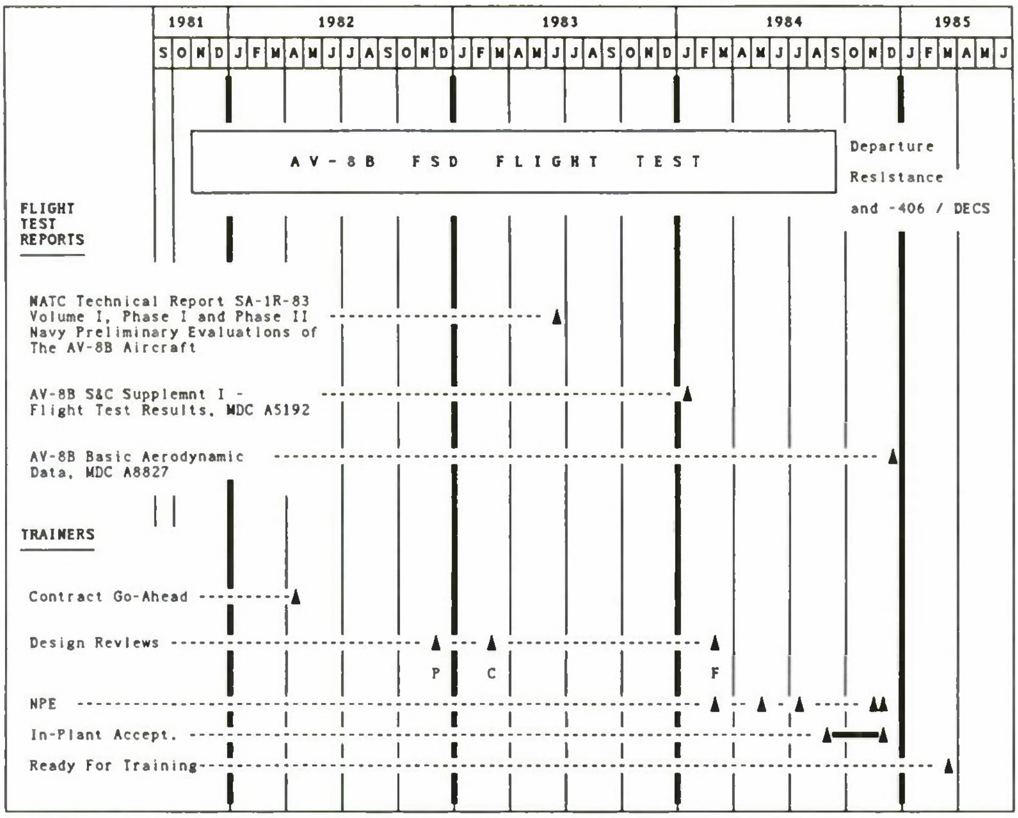


Figure 1. Flight Test Documentation and Trainer Schedule

SYSTEMS DESIGN

The various aircraft systems are normally documented by design reports that describe how the system is expected to work in the aircraft. These reports are usually the only information that is available initially to a trainer manufacturer attempting to develop a trainer concurrently with the aircraft. The level of detail in these reports satisfies the requirements placed upon the aircraft contract but usually is insufficient to develop the detailed models needed to accurately represent the systems. These reports describe what the systems will do, but not always how the systems will accomplish its purpose. Secondary effects on other systems are not detailed sufficiently to design the interaction of all systems.

Even if the trainer manufacturer had the details for a system per the design, these systems are also undergoing operational testing and being modified to overcome design oversights and problem areas. Thus, a trainer manufacturer may have modeled the original design perfectly but due to a later change the modeled system may be unrepresentative of the production system.

System changes to the aircraft typically fall into two classifications. The first classification is a Class I change. These changes have significant impact on the cost and/or design of the system and are well documented due to the need for customer approval for these types of changes. The second classification is a Class II change. These changes have less of an impact and are usually internally documented changes that the customer has given approval of without requiring the level of reviews for a Class I change. A Class II change is typically more difficult for someout outside of the aircraft project to track.

Changes made during the Full Scale Development effort typically do not require updates to the design reports that the trainer manufacturer is utilizing. Thus, it is difficult for a trainer manufacturer to be fully cognizant of the ongoing changes to the aircraft. Even if he was aware of a change, the detailed information needed may not exist or be readily available.

FLEET TEAM

The third issue is the lack of knowledge of the aircraft by the fleet user. A trainer manufacturer typically looks to the fleet project team to assist in refining the design of what is required on the trainer during design reviews. This can vary from the layout of the instructor station to the definition of the required malfunctions. Additionally, fleet project team evaluation of the trainer device prior to sell-off is a critical factor in the acceptance process for training systems. With no experience in the aircraft during the initial design phases for the trainer, the fleet team is limited in what they are able to interject as requirements for the system. With limited, if any, experience during the acceptance phase, many of the less significant items are overlooked. These factors placed a greater responsibility upon the trainer manufacturer to be more fully cognizant of the

needs of the customer and the total operation of the aircraft. Many of the questions that the fleet team previously answered are now required to be asked and answered by the trainer manufacturer himself.

These three issues come together to confront the trainer manufacturer with two significant items. First, the published data and customer knowledge available to him during the initial design phases of the program is insufficient to be able to design, develop and verify the trainer device. Secondly, the configuration of the aircraft is undergoing significant changes from what the trainer manufacturer was initially provided and thus a greater effort must be expended by the trainer manufacturer in order to perform the configuration management that previously may not have been required.

SOLUTIONS

The problem of concurrent development is to design the ATD from the approach that the aircrew trainers will be delivered in as close a configuration to the production aircraft as is feasibly possible. This philosophy has to be an integral part in the design of the aircrew training system. The initial design can be based on existing developmental simulation data and on published design reports for the various aircraft systems. Further clarification of the systems operations can be provided by asking specific questions of the cognizant design engineers from the aircraft company. With this information, the initial trainer system is designed.

The design and verification of the handling qualities model is based around the published flight test schedule. The basic flying qualities are tested first and performance testing is delayed until adequate criteria is available. Typically, performance testing follows behind the handling qualities flight testing. Additionally, the published results of flight testing lag the actual testing by many months. As an example, on the AV-8B, preliminary versions of the flight test results were provided by both Naval Aircraft Test Center (NATC) and the aircraft prime engineers. This permitted development of criteria and testing of the system before the data was actually published. The format of the data published by the aircraft company was changed by project engineers to more easily accommodate the requirements of the training device. By working with the aircraft engineers, it is sometimes possible to have the data provided in a format that is more meaningful and easier to use in the trainer community.

Problems will undoubtedly be encountered on the aircraft during a FSD program. The aircraft design staff will be required to devote their full attention to resolving these problems as they are uncovered. During these critical points in time, some of the data needed to verify the trainer is not available in an analyzed format when it was required on the trainer program. This results in the trainer designers needing to acquire raw flight test data (time histories and/or tabular data) identified by cognizant NATC and aircraft engineers and analyzing the data themselves. In order to reduce this data, additional information is often required from the aircraft engineers identifying the specific configuration of the aircraft (i.e. flap system schedule, control system revision level, any modification from the production configuration, etc.) for the given flights. The notes that the fight test engineers take during the flight are also helpful to assist in developing an understanding of precisely what the data is indicating. For the AV-8B, this assistance from the project engineers was instrumental in allowing the data to be analyzed within the time constraints of the trainer project.

Thus, the ability to develop the handling qualities model and the criteria data to verify the model was the result of a teamwork effort by NATC engineers, the aircraft project engineers and the trainer engineers. Without the assistance of the project engineers, the required criteria data would not have been available when needed.

The modifications that are made to the various aircraft systems can be tracked through direct participation in the aircraft prime's Configuration Control Board (CCB) for aircraft and configuration management. This provides the trainer project the means to determine which changes to the configuration of the design basis aircraft are required on the trainer. Once the Technical Orders (T.O.s) began to be published, they can be utilized to refine the models of the various systems. Whenever confusion arises, the cognizant project engineer should be contacted for clarification on how the actual system functions.

Again, the ability to contact the cognizant engineers on the aircraft is a significant item in being able to develop the trainer in a production configuration.

The third issue, concerning the fleet team, is the most difficult to resolve. The amount of knowledge available to the fleet team during the design phase of the trainer is no greater than what is known to the trainer design engineers. Specifics of what the fleet considers as needed on the trainer cannot be provided during the design phase of the trainer. The initial malfunction list for the AV-8B ATD was based on knowledge of the AV-8A aircraft. Several of the malfunctions listed in the original AV-8B trainer specification were not applicable due to system changes between the aircraft. The trainer project engineers must develop the design philosophies based on inputs from the fleet team, the contracting agency and cognizant test pilots. Several assumptions and designs must be made by the trainer designers to supplement the limited knowledge available from the fleet team.

Experience on the AV-8B training devices has revealed that some of the approaches taken were less than optimum. Comments from the fleet on several minor issues have identified these areas and they are being investigated by the trainer support activity responsible for the trainers. Most of these issues would have been discovered and corrected had the fleet team had more experience in the aircraft before acceptance testing of the ATD was performed. However, with concurrent trainer and aircraft development, this will never be possible and these types of problems must be expected.

Overall, the AV-8B trainer system has been widely accepted by the fleet users. One of the reasons for this overall acceptance is due to several of the trainer engineers having worked on

the AV-8B developmental simulation and having a very strong insight into what the Marine Corps was looking for in their aircraft. This helped ease some of the trainer designer decisions. Once again, being closely involved with the aircraft development assisted in overcoming a critical point on the development of the trainer.

OTHER CONSIDERATIONS

Two other considerations must be presented when discussing concurrent trainer and aircraft development. First, the contractual vehicles that aircraft and trainers are procured under cause long delays in the upgrading of the ATDs in order to maintain configuration. On the aircraft project, the contract is designed such that changes can occur in a timely manner. Once the aircraft change has been identified, it is a fairly lengthy aircraft change into the trainer. During concurrent ATD and aircraft development, this can lead to a trainer being delivered that is out of configuration with the aircraft.

The second consideration is attempting to alleviate this problem by having the aircraft prime be the procuring agency for the trainer devices. In theory, this approach should solve most of the problems with concurrent trainer and aircraft development. However, if the aircraft prime and the trainer contractor do not form a good interface at the worker level, this approach will not provide the solution to the problem. Ideally, the aircraft prime must consider the trainer with as much importance as it does the aircraft itself. The prime must provide the resources needed to enable the trainer contractor to adequately develop the trainer in a timely manner. The trainer contractor must be responsive in recognizing the aircraft manufacturers primary goal of developing an aircraft. The aircraft prime and trainer contractor must have a cooperative agreement with clear goals and understanding by all involved.

SUMMARY

There are several issues that make concurrent trainer and aircraft development very difficult. These issues can be resolved only if a working relationship exists between the aircraft prime and the trainer contractor. Whether the approach taken is that the aircraft prime is the procuring agency for the ATD or another approach is used, teaming by the aircraft prime and the trainer contractor must occur to successfully develop an aircrew trainer device on schedule and in the configuration of the design basis aircraft.

REFERENCES

1. Specification for AV-8B Operational Flight Trainer Device 2F133, NTEC Specification 222-1197A, 19 January 1984

2. Smith, 1Lt. Barry R., "SIMULATOR DESIGN CRITERIA: NEW EMPHASIS IN CONCURRENT DEVELOPMENTS", IITSC Proceedings, Volume I, 18-20 November 1986

ABOUT THE AUTHOR

Mr. Michael Frankie is a Software Engineer for the McAir Training Systems. He is currently the software manager on the AV-8B Aircrew Trainer Program. He was the lead engineer for the development of the flying qualities and aircraft systems for the initial AV-8B Aircrew Trainers. Prior to the Trainer program, Mr. Frankie worked on various engineering simulation programs in the Flight Simulation Department of McDonnell Aircraft. Prior to his employment by McDonnell Aircraft, Mr. Frankie worked for the Link Division of Singer in Binghamton, New York. Mr. Frankie holds a Bachelor of Science Degree in Aerospace Engineering from Parks College of St. Louis University.

MIL-STD-1379: DEVELOPMENT OF A "SINGLE" STANDARD FOR CONTRACT TRAINING

Authors: Dr. Leonard Courchesne
Assistant Deputy for Weapons Systems MP&T Programs
Naval Air Systems Command
(W) (202) 692-2137 (H) (703) 243-1233

Dr. Richard Eddings
Manpower, Personnel, and Training Division
Naval Sea System Command
(W) (202) 433-5742 (H) (703) 569-0270

Abstract for 9th I/ITSC

This paper discusses recent developments in the evolution of Department of Defense Standard 1379 (series) and associated data item descriptions. The standard is used contractually to state requirements and the data item descriptions provide industry with format and content requirements for the preparation and delivery of training materials.

The military services previously have used a myriad of source documents, loosely developed around a nebulous instructional systems model, to contract for development of training materials. Several inspector generals have stated that the instructional systems development process is too expensive and too subjective. MIL-STD-1379 (series), the Department of Defense "single" standard for development of training materials, recently has been promulgated in a new, interim revision, MIL-STD-001379C, for use by the Navy. The emphasis of this paper is on the usefulness of the latest standard. This new single standard and its successor, DOD-STD-1379D, should reduce the conflicts, redundancies, omissions, and inefficiencies of previous military training materials.

INTRODUCTION

In 1982 the Navy Training community emphasized a need for standardization of course curriculum requirements. At that time the community stated that training materials developed for use in Navy training were too expensive and were either procured or developed from too many different source documents. The use of several procurement documents created problems, such as conflicting terminology and anomalous formats, which were not only confusing to Navy training personnel but also frustrating to development contractors as well.

Perpetuation of more than one training materials development system has been a continuing problem in the training community and has inhibited effective communication of training ideas and materials. Lack of standardization of training materials has contributed to increased production costs and has magnified the administrative problems because of standards and procedures being prescribed on a case by case basis. Also, the monitoring functions have become more complex since DoD personnel were required to be knowledgeable of several differing standards and procedures. The difficulty in training instructors, changing curricula, and effectively using curricula developed at different training activities all focused on the need for a single standard. This single standard would have to meet the requirements of all DoD training communities, and it would have to utilize the best features of each of the existing systems.

In 1983 a working group was established to review existing standards and provide recommendations for development of a single standard. The outcome of this meeting was a "strawman" for a single military standard "MIL-STD-001379C" Military Training Programs and associated Data Item Descriptions (DIDs). It was also recommended that a "handbook" be developed to provide procedures and examples for development of products identified in the DIDs. This handbook would be used by both contractors and in-house activities. It was also concluded during the meeting that more emphasis should be placed on use of the NAVSEA OD 45519, Submarine Training Material Development Guide and Production Specification, and its associated management documentation.

In 1984 the Chief of Naval Operations directed the surface warfare community to use NAVSEA OD 45519 and related MIL-STD- 1379B Contract Training Programs Data Item Descriptions as the single standard for procurement of training courses and materials. During this period work also was underway to develop and promulgate a single military standard to establish requirements for the preparation, conduct, validation and verification of military training programs.

In 1985 MIL-STD-001379C (Navy), Military Training Programs and associated Data Item Descriptions were approved for use in lieu of MIL-STD 1379B. In addition, MIL-STD-001379C (Navy) superseded NAVSEA OD 45519. A related document, DOD-HDBK-292, was

conceptualized at this time and, with developmental assistance from the Chief of Naval Technical Training staff, was approved in 1986.

In 1986 work began on development of Department of Defense (DOD) standard, DOD-STD-1379D, Military Training Programs and associated DIDS, to support the training program requirements of all military services. DOD-STD-1379D will supersede MIL-STD-1577A (USAF), MIL-STD-1379B, MIL-STD-001379C (Navy) and MIL-T- 29053B(TD). DOD-HDBK-292(SERIES), as an accompanying guide, is designed to support all components of the Department of Defense.

DOD-STD-1379D

This standard and associated DIDs establish the requirements for the preparation of training materials to be used in the development and conduct of military training programs. This standard is applicable to all military training programs when invoked by contract.

This standard retains all of the good elements of other standards. For example, it stresses the performance of job related tasks in a practical/laboratory environment. Classroom/lecture time is held to a minimum. Training programs are designed and developed such that the Government may use the program to perform any future training.

A systematic approach to training is used to develop the training program and training materials. This approach involves control and coordination and integrates the systematic phases of analysis, design, development, implementation, and evaluation (see figure 1).

Phase I: Analyze

The first step in this systematic approach is to analyze the job (see Figure 2). The training process, as outlined in DOD-STD-1379D, begins with a planning conference. The DIDs used for the conference agenda and minutes are generic and were developed by the United States Air Force (USAF).

The training conference is an integral part of the preparations and acquisition of the training program. The conference will enable all parties to reach agreement on details of the training program before it is established.

The Development of a training program and training equipment plan (TP/TEP) in DOD-STD-1379D provides the Government with information concerning the contractor's plan for developing training.

The TP/TEP will be developed to provide information relative to the production of a training program needed to train personnel to install, operate, control, maintain, or otherwise use hardware or perform non-hardware related tasks or functions. It shall include a training project plan, course schedule data, and required supporting data.

The training program and training equipment plan identifies the recommended participants, locations, instructional materials, instructional methods, and time schedules of the training program. The course schedule data is used to plan, develop, and monitor the materials, events, and personnel required to implement the training program.

By using the standards set forth in DOD-STD-1379D and the guidelines in the handbook (DOD-HDBK-292 (Series), training programs and training equipment plans can be established which fully meet the needs for the military communities and provides standardization that will encourage an exchange of training ideas.

The Manpower, Personnel and Training Analysis of DOD-STD-1379D combines, in a systematic process, job analysis, selection of tasks, and job performance measures. This analysis is conducted to accurately identify tasks which will be performed by an operator, controller, maintainer, or other personnel. Included in the manpower, personnel, and training analysis is the training analysis data sheet (TADS) which provides an analysis of the knowledge or skill steps of a training task. It hierarchically identifies the steps or substeps in relation to the training task/functions as they apply to skill levels. Such information is then used in the development of a training analysis task summary. The summary lists by task, all steps, substeps, skills and knowledge and is arranged in a matrix format. This information is used to develop the training analysis task list which summarizes, in matrix format, all tasks, which are selected for training.

The training analysis task list is a control document providing a composite picture of all training task requirements. It includes behaviors, conditions, and standards of performance.

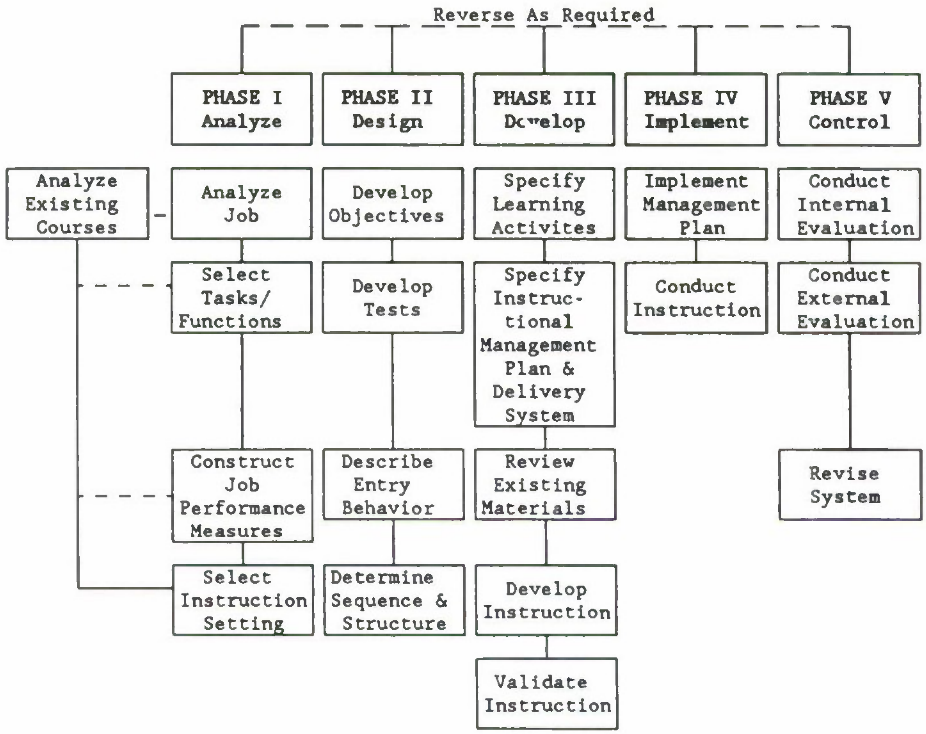
The manpower planning data include the personnel requirements for operating and maintaining a system, subsystem, or equipment. The summary provides quantitative manpower requirements by personnel specialties and skill levels.

Personnel performance profiles and the training system are subsets of the analysis phase which are used to define the minimum requirements (knowledge and skills) to operate and maintain a system, subsystem, or equipment or perform tasks/functions.

Phase II: Design

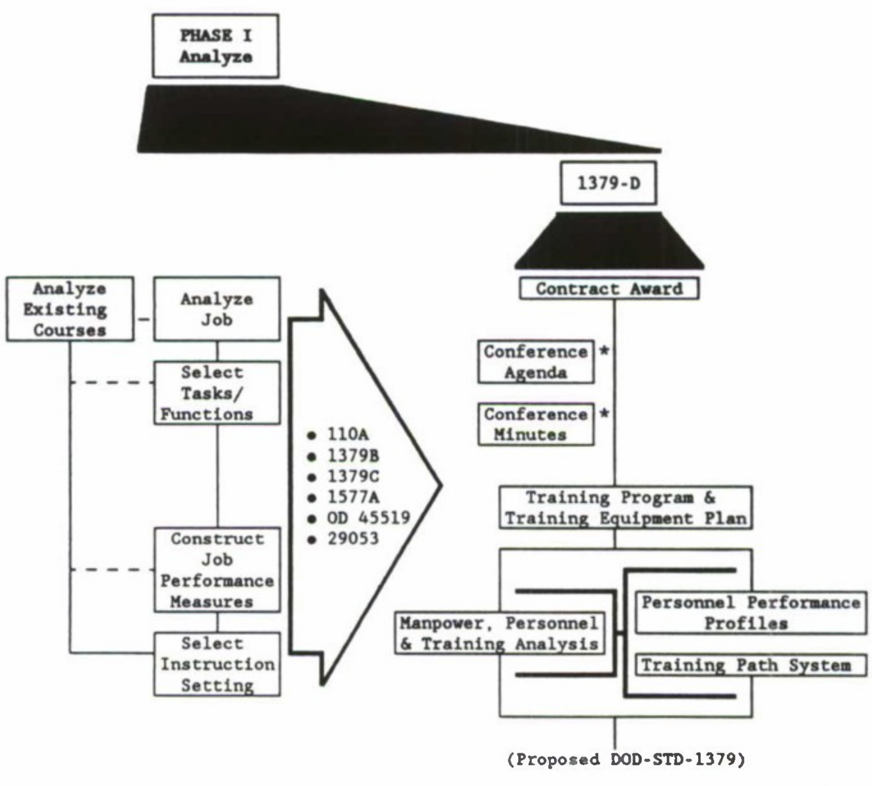
Designing the training program involves conversion of tasks into objectives, determination of test items, sequencing of information to be taught, and selection of the best methods and media required to support the transfer of knowledge and skills (see Figure 3).

The first element is the development of objectives. Once the task analysis is complete and course objectives set, a Topical Outline will be developed. The topical outline identifies the goals of the specific course and order of subject matter presentation. Course Learning Objectives (CLO) are provided in the topical outline. Course Learning Objectives describe the overall knowledge and skills to be attained upon completion of the training. Development of CLOs in turn facilitates the development of Topic Learning Objectives. TLOs consist of a list of all learning objectives for a specific training path. The learning objectives are prepared to reflect the coverage that is to be provided in the individual topics. Learning objectives contain three essential parts: Behavior, Conditions, and Standards and serve as a "common thread" which identifies specific job related tasks. To ensure that the attainment of



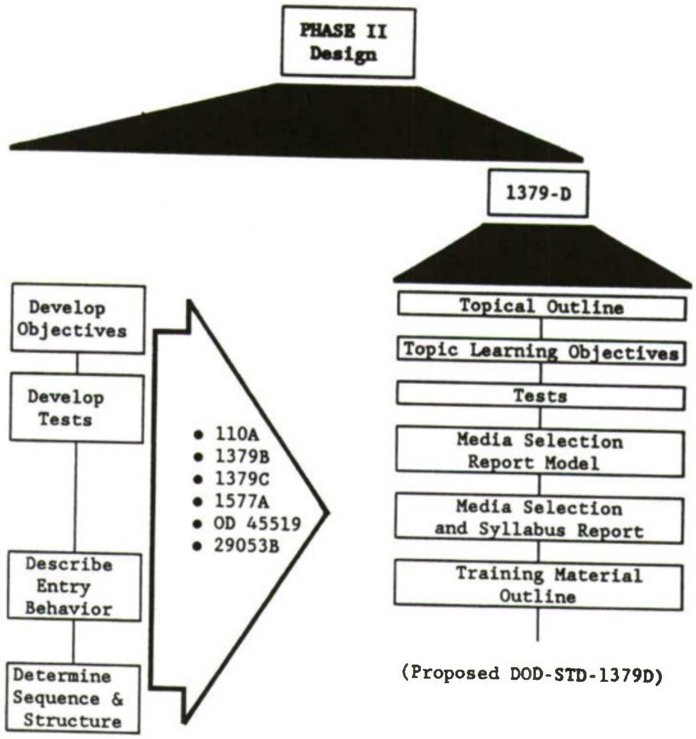
I S D MODEL FLOWCHART

FIGURE 1



ISD PHASE I and proposed DOD-STD-1379D

FIGURE 2



ISD Phase II and proposed DOD-STD-1379D)

FIGURE 3

learning objectives are met, tests for measurement of trainee achievement must be developed.

The second element is development of tests. Two types of tests are developed along with a proctor guide for each type. A knowledge test booklet provides a means to measure trainee achievement of learning objectives. A performance test booklet is used to measure trainee performance during a laboratory exercise.

The third element of the Design Phase is determining sequence and structure. Proper sequencing will help the trainee make the transition from one skill or body of knowledge to another and will assure that the supporting knowledge and skills are acquired before dependent subject matter is introduced.

The final element is the development of a method and media model which details the procedures to be used in selecting primary and alternate media for all objectives. Then, each learning objective is exercised through the model to determine all media required to support the learning objective.

The Training Material Outline (TMO) provides detailed recommendations with justification for instructional media material, in relation to each topic listed on the outline. When approved, the TMO becomes the master plan for production of IMM. These include transparencies and wall charts, story board- scripts, sound-slide presentations, videotapes, or motion pictures.

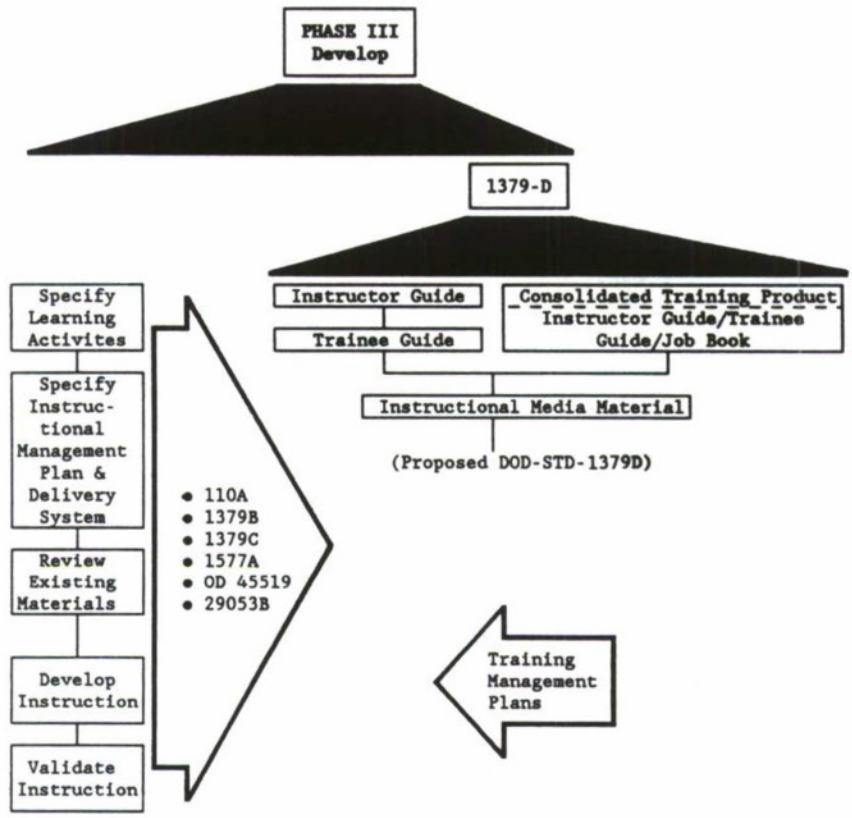
Phase III: Develop

Development of training materials involves the systematic completion of draft training materials and the validation and verification of final training materials. (See Figure 4)

The instructor guide (IG) provides direction concerning training objectives, equipment and instructional media material use, and conducting the course. The instructor guide should contain sufficient detail to ensure that the proper depth of coverage is achieved consistently.

The trainee guide (TG) is developed to enhance the trainee's acquisition of the knowledge and skills. The trainee guide normally includes instruction sheets which consist of assignment, diagram, information, job, and problem sheets.

DOD-STD-1379D also provides for the Consolidated Training Product (CTP) which was developed based on the U.S. Army Soldier's Training Product methodology. The consolidated training product integrates the Instructor Guide, Trainee Guide and Job Book under one cover. CTPs contain standardized training objectives in the form of task summarizes which identify the individual requirements for trainees in specific occupational specialties or for common skills. CTPs are designed for use by commanders and instructors to plan, conduct, and evaluate individual training in units.



ISD Phase III and proposed DOD-STD-1379D

FIGURE 4

Formal training may be supplemented by development of any of the following products as identified in the training material outline:

An Exercise Controller Guide (ECG) - a set of exercises for use either in formal of informal training environments.

Lecture Guides (LG) - an outline of major sections, key topics, and discussion points, used with other instructional media such as slide programs, video tapes, or motion pictures.

Self-Study Workbooks (SSWB) - books for self-learning which may be used in conjunction with an administrator guide to provide a controlled path of self-study for specific skill tasks.

On-the-job Training (OJT) Trainee and Instructor guides (IG) - guides which may be developed to supplement or replace formal training at the trainee's work site.

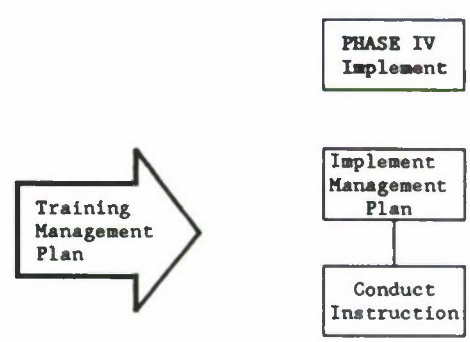
Alternatives to formal training are:

Technical Hands-on Training System (THOTS) Trainee and instructor guides - materials that provide a self-paced form of instruction designed to teach operation and maintenance skills instead of using formal stand-up instruction.

Simulation User's Handbook (SUH) and Simulator Software Handbook (SSH) - materials that are designed for use by simulator/stimulator, operators and maintainers.

Phase IV: Implementation

Implementation of training programs is supported by implementation and management plans. The implementation and management plans identifies procedures and covers problems unique to specific instruction, and actually brings the instruction on-line. (see Figure 5)



ISD Phase IV and proposed DOD-STD-1379D

FIGURE 5

Phase V: Evaluate

The evaluation phase deals with procedures and techniques for maintaining instructional quality control and for providing data from internal and external sources upon which revision decisions can be based. Data collection, evaluation of the data, and decision making about the implications of the data represent the three principal functions. Emphasis is placed on the importance of determining whether the trainees are learning what was intended and upon determining whether what they have learned is of the expected benefit to the receiving command. (see Figure 6)

CONCLUSIONS

One way to indicate the difference between DOD-STD-1379D and existing military standards, and the Instructional System Development processes is to point out that there are currently a number of existing standards and one Instructional System Development process. Some of the standards represent excellent applications of the systematic approach to training. There are outstanding examples of well-conceived and delivered instructions available within the interservice training community. A major purpose of this standard is to establish a single interservice military standard for the design, development and delivery of instruction which will meet state- of-the-art specifications.

This standard requires the application of one version of the Interservice Procedures for Instructional System Development, by referencing available methods to the fullest degree possible in order to optimize training effectiveness, efficiency, and cost. This military standard is intended to be used in the development of military training programs, related data, and training materials. The recommended content of DOD- STD-1379D will aid in the development of consistent quality material for use throughout the Department of Defense (See Figure 7).

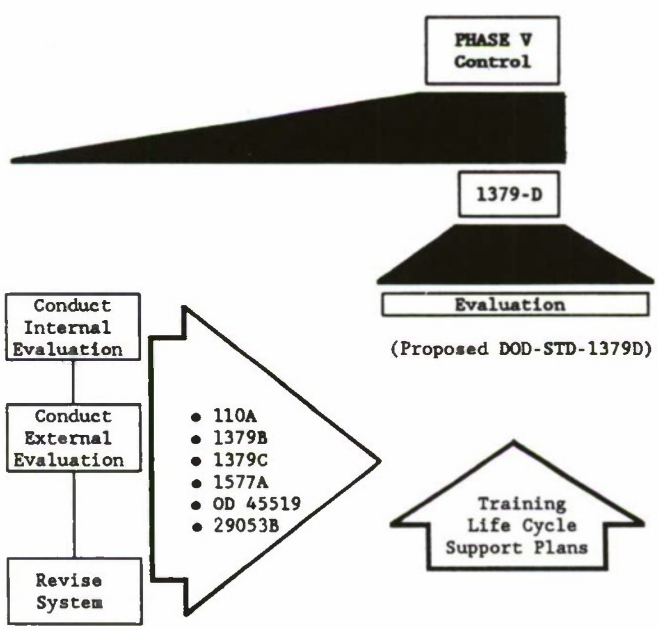
Finally, when this military standard is used in an acquisition an data are required to be delivered, the recommended data requirements shall be developed as specified in the contract.

REFERENCE

* LDR ltr ser 86-160 of 20 August 1986
  LD Research, Falls Church, VA

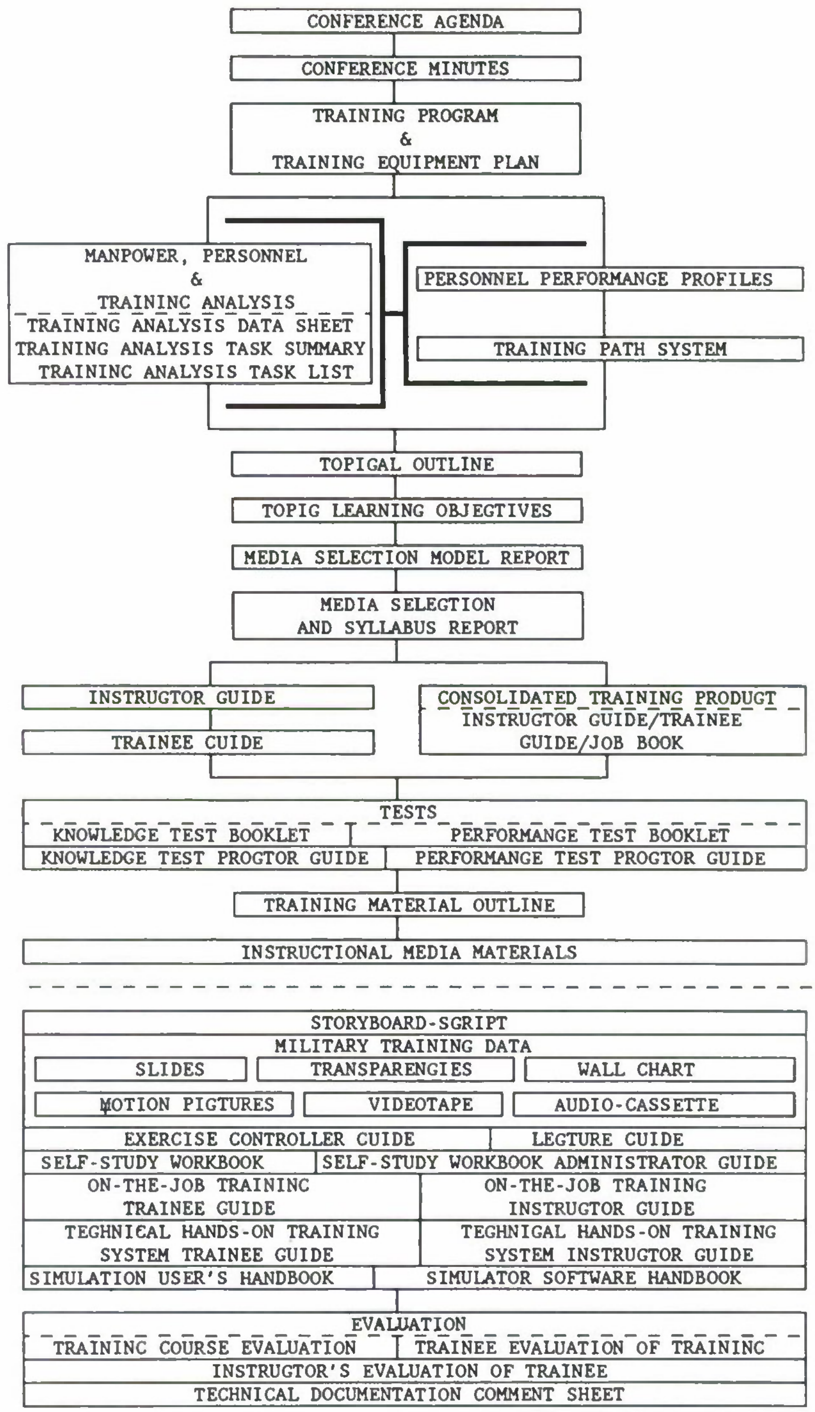
* MIL-STD-962A, Military Standards and Handbooks, Preparation of

* NAVEDTRA 106A Interservice Procedures for ISD



ISD Phase V and proposed DOD-STD-1379D

FIGURE 6



Proposed DOD-STD-1379D Products

FIGURE 7

ABOUT THE AUTHORS

Dr. Leonard Courchesne is an instructional technologist assigned as Assistant Deputy for Weapons Systems Manpower, Personnel and Training Programs, Aviation Training Systems Program Office, Naval Air Systems Command. He has used all series of Military Standard 1379 in obtaining contract training for the Navy, Army and Air force. He has trained bedouins, infantry soldiers, Army field grade officers of Command and General Staff College, sailors in aircraft maintenance, Naval aviators, airmen, Air Force lieutenants at Officer Training School, Strategic Air Command pilots and bombardiers in RB-47 and FB-111, and civil service engineers in reliability, maintainability and quality assurance.

(work) (202)-692-2137
(home) (703)-243-1233

Dr. Richard Eddings is a training specialist with the Manpower, Personnel, and Training Division, Deputy Chief Engineer for Logistics, Naval Sea Systems Command (NAVSEA). He is the Navy lead service coordinator for Military Standard 1379. Dr. Eddings participated in the early decisions for the Navy to adopt a single standard for procurement of training materials and was the NAVSEA representative throughout the development of MIL-STD-001379C (Navy). He was assigned the responsibility for MIL-STD-1379 in 1985, when the Chief of Naval Material was disestablished and functional responsibility for the standard was transferred to NAVSEA. His current efforts include DOD-HBK- 292, Training Materials Development, approved 31 October 1986, and development of DOD-STD-1379D for tri-services use.

(work) (202)-433-5742
(home) (703)-569-0270

SOFTWARE DOCUMENTATION ON MAGNETIC MEDIA AND THE TRAINER COMPUTATIONAL SYSTEM

Steven R. Prince
AAI Corporation
P.O. Box 126
Hunt Valley, Maryland 21030-0126

ABSTRACT

As trainer systems become more complex, the amount of software required to implement these systems increases. Consequently, the amount of documentation necessary to support the trainer software also increases. It is now typical for initial trainer system documentation to number over 50,000 pages. Over the life of the system, due to change activity and resubmittals, this number can increase ten-fold. At the outset of the EF-111A Operational Flight Trainer (OFT) Program, the heavy burden that this paper volume could place on both contractor and customer was recognized.

A suggestion was made to maintain the EF-111A OFT software documentation on the trainer computational system and deliver the documentation to the customer on magnetic media. Both the customer and contractor determined that potential cost savings as well as capabilities not available under the present paper system were attainable. Consequently, a change was incorporated into the contract to allow magnetic media delivery of software documentation, and modifications were made to documentation formats to accommodate its use. The change was implemented with no impact on the program schedule and in such a way as to minimize the need for new tools, software, or hardware.

This paper describes the system that was developed to provide the creation, maintenance, and delivery by magnetic media of the software documentation on the trainer computational system. The lessons learned, problems encountered, and successes realized from the effort are detailed. Topics include methods used for documenting changes resulting from software element changes, incorporating subcontractor documentation, providing text editor capabilities to the customer, incorporating source file data into documents, interrogating configuration management files to determine software and documentation status, handling illustrations and special characters, and maximizing resources.

INTRODUCTION

EF-111A OFT Software Overview

The EF-111A OFT is an operational flight trainer that provides realistic simulation of the aircraft tactical jamming system and the electromagnetic environment in which the aircraft operates. The trainer consists of four software subsystems as well as a crew station and instructor operator station. The four software subsystems are the flight, tactics, radar, and development (DEPS). AAI, the prime contractor on the program, developed the tactics and DEPS subsystems while subcontractors developed the radar and flight.

The EF-111A OFT software structure is depicted on Figure 1. The tactics subsystem, presented in detail, is typical of all four subsystems with the exception of the DEPS, which does not include real-time software. The software structure of Computer Program System (CPS), subsystem, functional area, Computer Program Component (CPC), Computer Program Module (CPM), and software element is established per EF-111A OFT contract requirements. CPM's and software elements, the two lowest levels of the software, are not completely depicted on Figure 1 since they consist of many members. The 6-digit numbers appearing on the illustration identify the components at each software level. Definitions of the software levels, as incorporated into the contract, are as follows:

CPS - the overall EF-111A OFT system

Subsystem - major system components

Functional Area - aggregates of specific functions (CPS's) that perform tasks that are logically related to fulfilling system requirements

CPS - a grouping of CPM's, which performs a major function

CPM - a FORTRAN subroutine called by the real-time executive; CPM and module are used interchangeably

Software Element - all subroutines, functions, job control streams, catalog command files, data files, etc., that will be delivered as part of the CPS

The software structure also distinguishes between the real-time and nonreal-time type of software. Real-time is defined as trainer simulation software used in the operation of the trainer while nonreal-time is all off-line support software.

EF-111A OFT Software Documentation Overview

The EF-111A OFT software documentation consists of a Computer Program Development Plan (CPDP), Computer Program Development Specification (CPDS), Computer Program Product Specification (CPPS), and Training Equipment Computer Program Documentation (TECPD). Each adheres to MIL-STD-483 and MIL-STD-490 requirements imposed on the program.

The CPDP is a planning document that portrays the methods used for developing software. The CPDS documents the early design and development of software. The bulk of EF-111A OFT documentation consists of the CPPS and TECPD documents. A CPPS volume is generated for the CPS, for each subsystem, and for each CPC of a subsystem. As can be seen from Figure 1, the portion of the CPPS describing the Tactics subsystem consists of 35 separate volumes; 1 at the subsystem level and 34 at the CPC level. The TECPD is divided into three volumes. The first volume contains programming and documentation standards, the second is made up of users guides (32 separate documents), and the third consists of vendor manuals.

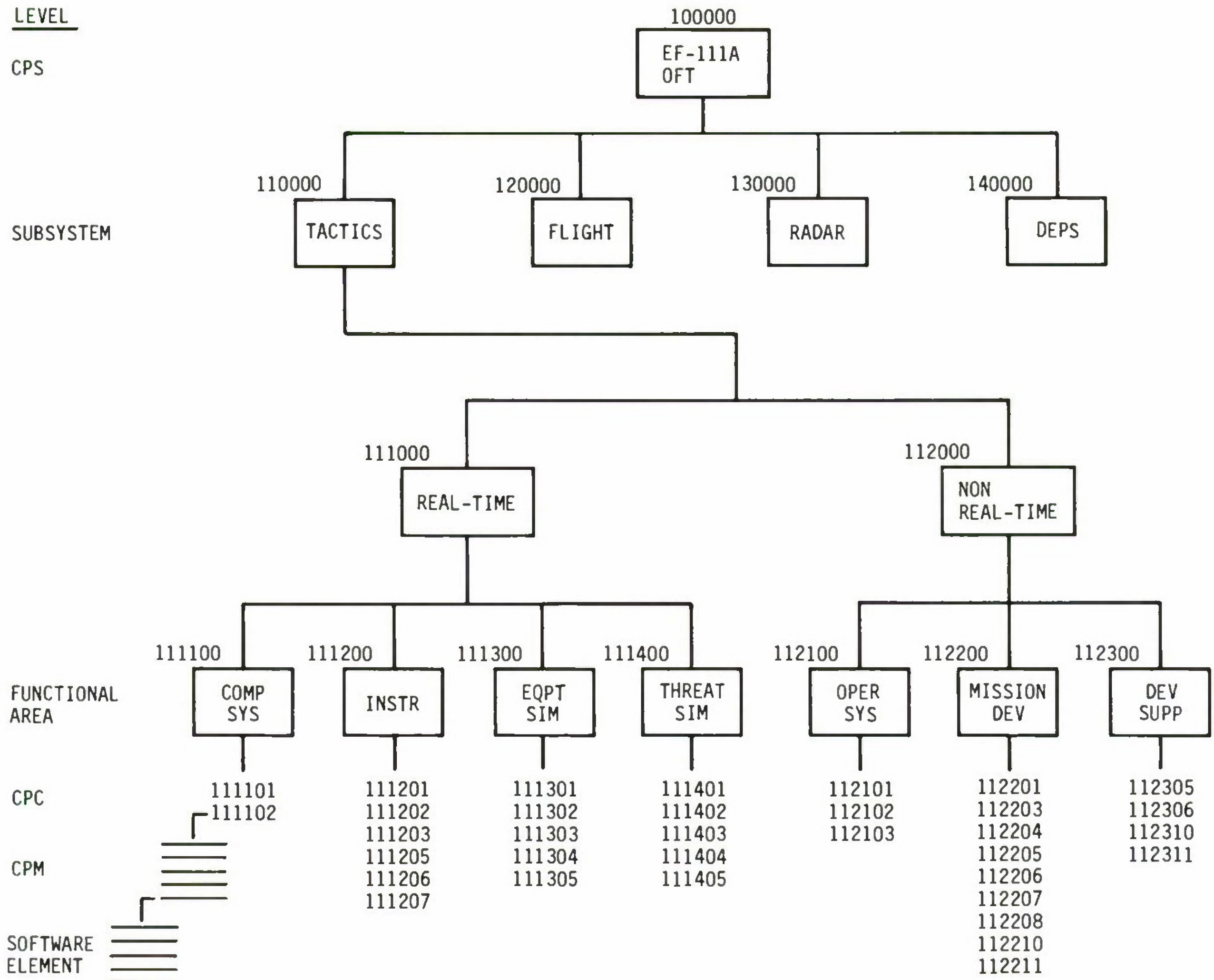


Figure 1. EF-111A OFT Software Structure

HISTORY

At the outset of the EF-111A OFT program, the customer and AAI agreed to investigate the feasibility of (1) delivering all software documentation via magnetic media as opposed to the standard method of multiple hard copies per document and (2) housing all software documentation on the development computer. ASD was looking for a new software documentation concept using magnetic media. In particular, a trial case or "test bed", which would allow ASD engineering personnel to monitor a contract for technical and functional development by reviewing software development products (documentation) produced by this new approach was desirable. Additional study into housing the magnetic media documentation on the system computer was desired in part due to an in-house ASD white paper that praised the virtues, and low life cycle costs, of embedded computer program documentation.

Magnetic Media

During the ensuing months, discussions and correspondence between ASD and AAI helped to define more clearly the goals and requirements of the proposed magnetic media approach. The following items were established as fundamental to the concept.

(1) Magnetic media deliveries should be accompanied by one paper copy submittal of each document, thus providing an easy method for comparison of the new approach while relieving both the Air Force and AAI of the burden to process and handle multiple paper copies.

(2) A minimum number of new tools or resources should be required to implement the new system; however, the user should have the ability to easily examine data files.

(3) The level of data content should meet CDRL requirements. Format of data items could be changed to accommodate magnetic media needs but technical content could not be compromised.

(4) The only software documents to be presented by magnetic media should be the CPPS's and the TECPD users guides.

(5) The concept should be required to mate with the SEL (GOULD computer system) environment with possible future VAX applications.

(6) The new method should provide ASD with a better monitoring system due to the ease of remote terminal documentation viewing. Consequently, the IV&V effort should decrease.

Embedded Documentation

Concurrent with the aforementioned magnetic media discussions, AAI conducted an internal cost study on embedding software documentation on the development computer for the EF-111A OFT program. The initial premise of the study was that software documentation does not just document the end product but rather the ongoing development effort. As a result, many things in the documents change and many updates/resubmittals are required. This was determined to be the main cost driver in software documentation, especially when identical information was in numerous places and required updating. The solution appeared to be embedded software documentation with primary value in data base files, code preambles, text files, and source code listings. The AAI study generated the following scenario for embedded software documentation.

Data describing variables is entered on disc via the system editor, thus creating a data base file. By restricting the ability to modify this file, configuration control over system design is possible as well as considerable design/mechanical error check capabilities. In-house developed programs interrogate the data base file for a number of applications. As a result:

(1) Module/variable cross-references are automatically produced from the central data base file. (This information was presented in the EF-111A OFT CPPS Subsystem volumes.)

(2) Input/output tables are automatically produced from the central data base file. (This information was presented in the EF-111A OFT CPPS CPC volumes.)

(3) The need for describing any item more than once (and possibly differently) is eliminated. When a variable description changes, it need only be changed once since recompiling all effected modules would automatically incorporate the change.

(4) The information in the data base file is all that is needed to write specification and initialization statements in the code for most system variables. Furthermore, they will be guaranteed to be correct.

All module type documentation is entered to the preambles of code in the form of headers. This includes Program Design Language (PDL), which could adequately portray the design in lieu of flowcharts. In addition, by using structured PDL, a PDL flowchart generator program can convert the PDL into a flowchart presentation. As a result:

(1) Accuracy of module descriptions and PDL is assured.

(2) Automatic incorporation of preamble information into documents eliminates manual generation of module descriptions and subsequent word processing of text. (This information was presented in the EF-111A OFT CPPS CPC volumes.)

Text files consist of author inputs as well as the code preambles, module/variable cross-references, and input/output tables previously mentioned. Both text files and source code listings are incorporated into documents via automated procedures directly from the system computer. As a result:

(1) Real-time status is achieved and documentation concurrency is assured.

(2) Word processing time is eliminated due to the ability to directly incorporate data generated from source code listings and data base files.

(3) Control procedures imposed on these files and listings assure that only approved changes are implemented and documented.

From the activities discussed, the approach for EF-111A OFT software documentation evolved. The remainder of this paper discusses the implementation of the approach and the results achieved. It will become clear that the ideas, goals, and findings of the studies and meetings previously discussed were incorporated into the EF-111A OFT program.

CONTRACT MODIFICATIONS

After the Air Force approved the magnetic media/embedded documentation approach, a number of contractual modifications were necessary. The Air Force modified the delivery requirements for the TECPD's and CPPS's to allow magnetic media delivery; however, as recommended by ASD, one paper copy was to accompany each magnetic media submittal. This allowed ASD comparison capabilities as previously discussed and caused AAI to assemble each document in paper form, thus ensuring the capability prior to Air Force delivery.

The data item descriptions (DID's) for the EF-111A OFT TECPD and CPPS were modified. The DID for Volume I of the TECPD, which provides standards and formats to be adhered to, was modified only slightly. Specifically, it was modified to allow portrayal of flowcharts on horizontal paper and to allow the flowcharts to be machine-generated from PDL comments. The CPPS had a greater number of modifications, although technical content was not compromised. A list of the effected CPPS sections and a brief description of the incorporated changes is provided in the following paragraphs.

Detailed CPM Narrative

- Allowed for a "narrative description taken from the preamble of the source file of the appropriate module or software element, and the PDL description which is also in the source file."

- Stated that the section would be "entirely machine generated."

- Excluded certain software elements, such as "control streams, catalog commands, data files, etc.," from requiring PDL and flowcharts.

Detailed Algorithm Modeling

- Specified that "this section must be manually generated."

Block Diagrams and Flowcharts

- Specified that "All CPCs shall be provided with functional block diagrams. All modules, subroutines, and functions shall have a flowchart generated from PDL statements."

- Further specified that flowcharts produced from PDL via the automated flowchart generator and residing on magnetic media would not require hard copy submittal; however, "block diagrams and flowcharts not producible directly from media shall be produced on hardcopy".

CPS Listings

- Provided specific requirements on comments, headers, variables, algorithm models, etc.

Along with CDRL and DID modifications, the selection of equipment was necessary before the documentation effort could begin. Additional GFE was provided to support the new approach. This included two terminals, extra disc packs, a UNIX type tool kit providing edit capabilities, a tape drive, and a disc drive.

IMPLEMENTATION

Documents and source code were submitted via disc or magnetic tape. The contents were loaded onto the customer's computer system. The disc packs and magnetic tape were then returned to AAI. This allowed for the reuse of the magnetic media while providing the customer access to each revision via their own computer.

Documents consisted of distinct text files for each section. In some instances, major sections containing large amounts of text were further segmented. The major narrative sections of each document were entered into the system as a unique file name conforming to the following structure:

W.tvvxxx

where

t = subsystem

vv = volume number (specifies CPPS or TECPD users guide volume)

xxx = section number (specifies which section within the volume)

For example, W.T10240 was the file name for the text contained in Volume 10, Section 2.4 of the tactics subsystem. Composites of text files for a particular subsystem and data item were archived into a unique file for that document.

Tables 1 and 2 illustrate the sections of the CPPS and TECPD that were assigned unique files. They reflect the xxx section numbers from above and appear in each applicable volume. The sections of these documents were generated as a result of the AAI/ASD preliminary EF-111A OFT discussions and the DID modifications to the contract previously described. Generation methods for particular sections reflect those specified in the DID changes.

Table 1. CPPS CPC Sample Section Assignment

** Section	xxx	Generation Method
Scope	100	Automated from source code
CPC Description	200	No text
Requirements	210	Automated from source code
CPC Program Description	220	Automated from source code and data base files
Malfunctions	230	Author inputs
Limitations	240	Author inputs
Algorithm Modeling	300*	Author inputs
Quality Assurance	400	Author inputs
Appendix A - List of Software Elements	A00	Automated from source code
Appendix B - Tree Listings of Software Elements	B00	Automated from source code
Appendix C - Appendix nn	Listings	Automated from source code

* In a limited number of cases, Section 3.0 has been divided as follows:

W.tvv301
W.tvv302
.
.
.
W.tvv3nn

where n is a 2-digit number

** Also includes files for Table of Contents and List of Effective Pages.

Table 2. TECPD Users Guide Sample Section Assignments

*Section	xxx	Generation Method
Introduction	100	Author inputs
Program Execution	200	Author inputs
Sample Execution	300	Author inputs
Preparation Instructions	400	Author inputs
Appendix A	A00	Author inputs
Appendix B	B00	Author inputs
.	.	.
.	.	.
.	.	.
Appendix n	n00	Author inputs

* Also includes files for Table of Contents and List of Effective Pages.

NOTE: Files generated via author inputs were subsequently placed under CM and produced on magnetic media via the automated file system.

Text Files

Text data was initially input into the appropriate designated file by the programmer responsible for that area of design. Upon completion of the text data entry to a particular file, the file was stored in a limited access documentation directory and the documentation coordinator was notified. A hard copy printout of the file was then generated and reviewed by cognizant personnel. The final red line hard copy, indicating all modifications occurring during the review and edit process, was then returned to the documentation coordinator. All modifications to the text were then made by the documentation coordinator. When the final version of the text was complete, embedded text commands were added, which generated specification header data, change and revision numbers, and other formatting data. Upon completion, the text file was added to the rest of the particular data item. Each data item text file was archived in the documentation directory as it was completed. The documentation coordinator was the only person with write privilege to the file.

Module Preambles

In addition to the manually generated text described above, narrative text could be pulled from the module preamble headers for incorporation into documents. In-house developed programs performed the transfer from source code to documentation text file. After the transfer was complete, the text was handled as any other text; i.e., embedded commands were added and the file was archived into the proper document file. The ability to copy from source code to text files, rather than using word processing personnel to retype already existing text, was a big advantage of embedded documentation.

Data Base Files

Module/variable cross-references and input/output tables were generated by the interrogation of data base files. Once generated, this information was incorporated into documents in the same manner as text files.

Listings

CPPS CPC volumes contained listings in addition to narrative text. Each listing appeared as a separate appendix to each volume. The first two appendices were reserved for the list of software elements and tree listing of software elements for the particular CPC. Two in-house developed programs generated these appendices. Embedded commands within the programs provided format data such as specification header and date information to appear on the printouts. To obtain the actual module listings for a CPC (the remaining appendices), a number of internally developed programs were used. The programs (1) obtained CPC numbers for each routine from CM files, (2) scanned data files to obtain appendix letters for each routine within a CPC, (3) sorted and prepared appendix listings by CPC number, then by letter within each CPC, (4) built release files with specified formats such as headers, dates, etc, and (5) released, copied, or printed routines for the entire system, a specified subsystem, or a CPC.

Disc Usage

At this point, it is important to note the discs involved in the documentation process. Four types of discs were used: a user disc, a system disc, controlled subsystem discs, and a publications disc (Figure 2). A user disc, the first disc involved in the process, contained in the limited access documentation directory previously discussed. This directory contained the archived text files generated from author inputs, source code preambles, and data base files. The system disc, the second type involved in the process, contained the data base files used to generate cross-reference and input/output information.

The third type of disc used in the process was a controlled subsystem disc. There were a total of three subsystem discs, with the tactics and DEPS subsystems on one disc and the flight and radar subsystems on the other two. The subsystem discs contained two directories: one for source code and text and the other for listings. As part of the documentation process, all archived text files in the documentation directory of the user disc were copied into the text directory of the subsystem discs. Since listings were already resident on the subsystem discs, all the pieces of a particular document were now on the subsystem disc and under configuration control. The task now was to transfer both types of information to the publications disc, the fourth disc involved in the process.

The publications disc had directories for each type of document; i.e., TECPD and CPPS. In the case of the CPPS, the directory was further partitioned to distinguish between text files and listings (the TECPD had no listings). Text was copied from the

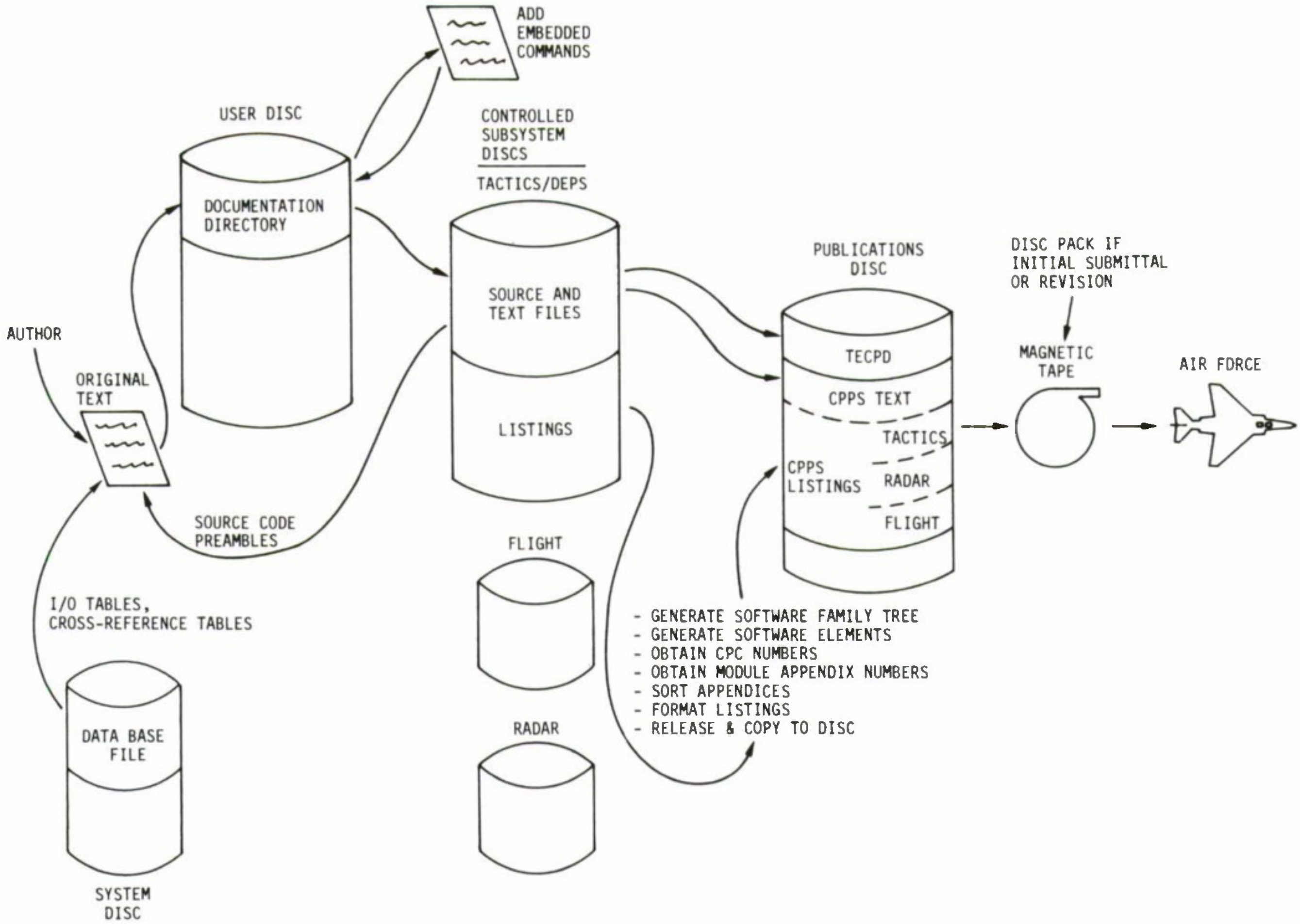


Figure 2. EF-111A OFT Documentation Process

subsystem disc to the appropriate text directory of the publications disc via a simple one-line command. The procedures previously discussed for formatting, sorting, and generating the CPPS listing appendices were implemented from the subsystem discs. The last step in the process released and copied the formatted listings from the subsystem disc to the publications disc. With both text and listings now on the publications disc, a simple command specifying the desired volume would combine the text and listing files for that volume into the single document being produced and copy the contents onto the magnetic media. The magnetic media, which was either a disc or magnetic tape depending on the type of submittal, was then delivered to the Air Force.

By using the above procedures, rigid control over the documentation process was achieved due to (1) limited access to the documentation directory of the user disc, (2) configuration control over the listings and files on the subsystem discs, and (3) a separate publications disc for final deliverable data. The document directories on the publications pack were cleared every month. After that, only documents that were to be resubmitted were put on the publications pack. When copying to magnetic media, it was possible via a single command to copy all new documents at once rather than specifying each individual volume.

AIR FORCE USAGE

After the magnetic media was delivered to the Air Force and copied onto the system computer, the Air Force was ready to review the documents. A Magnetic Media Users Guide, an additional deliverable document resulting from the magnetic media approach, assisted the Air Force in file usage. The document contained a list of all file names delivered in a particular submittal as well as usage instructions. Since access to the files was "read only", there were two ways this could be accomplished. A file could be viewed on a CRT or printed out as hard copy. The delivered files had embedded runoff and text commands throughout the text. It was possible to view or print out either a clean formatted version of the document (one without the embedded commands) or one containing these commands. The formatted version was identical to the paper copy version of the document submitted per contract requirements. The version that had the embedded commands, however, was difficult to read and offered no obvious advantage. Viewing or printing out either version was accomplished by no more than two simple commands.

When viewing a file on the CRT, the first 23 lines of the document would appear. When the RETURN key was pressed, the next 23 lines were presented, allowing the user to view the entire file with

minimum instruction. Limitations were that the user could only advance 1 screen (23 lines) at a time and there was no backward scolling; however, since documents consisted of many distinct files and each individual file could be viewed separately, the impact of these limitations was minimal.

Files could be printed out by either creating an output file or by using a spooling technique. The advantage to using an output file was that multiple copies could be produced via a single command. Implementing this method required additional disc space. Spooling required no additional disc space; however, the user had to repeat the spooling commands each time that a hard copy of the same file was required. The particular need at the time determined the appropriate method.

The advantage to both the CRT viewing or printout method of reviewing documents was obvious. The only limit on the number of people viewing the document was simply the number of remote terminals or line printers available with access to the system computer. Reproduction costs, turnaround time, and general paper shuffling were limited to whatever level the customer desired. This advantage grew in importance when subsequent submittals and revisions occurred. Changes could be easily incorporated and the most recent revision level of any section could be viewed from the system. Further, only desired portions of the documents (selected files) need be accessed.

RESULTS/FEATURES

Software Element Changes

When a revision to a CPPS or TECPD was required, a new disc and hard copy were submitted by AAI, superseding all documentation existing from previous submittals of the data item. In essence, this was a resubmittal of the complete document.

When a change to a CPPS or TECPD was required, a magnetic tape and hard copy were submitted by AAI. All changes were sectional changes; i.e., the entire section in which the change occurred was submitted. However, sections that did not contain changes were not submitted. Changes within each section were noted as follows:

#RC change to text RC#

where

R = current revision level

C = current change level

#RC = beginning of text change

RC# = end of text change

This was the magnetic media equivalent of the change bar used on hard copies; however, an advantage to this method was that changes could be automatically found by simply searching for the # symbol in files directly on the system rather than searching for change bars on paper copies. All #RC and RC# notations were used solely for change submittals and were removed with each revision.

When a change submittal was required, both text and listings could be affected. Text, already archived and under configuration control, was modified in a manner similar to code; i.e., text files were reserved from CM, edited, verified, and returned to CM control by the documentation coordinator. Listing changes were already incorporated into the code by the responsible engineer. At this point, all changes were in the text files and source files. However, only changed files were to be submitted; therefore, the task was to determine only those files that were changed. In the case of text files, this translated into document sections. In the case of listings, this translated into document appendices. Configuration management (CM) files were searched to determine text and source files effected by changes since the previous submittal of a document. A number of in-house developed programs were used to accomplish this. When determined, the changed files were copied from the subsystem pack to the publications pack for magnetic media transfer as previously described.

When changes were required, only the effected portions (i.e., files) were modified and resubmitted; however, the customer could regenerate the entire document using appropriate text files and commands. File manipulation capabilities allowed viewing or printing of countless combinations of sections if desired. Other benefits realized by the system were that real-time status was achieved since both CM and text files were on the development computer. Also, production time for hard copy text changes was eliminated. Incentive fees for keeping documentation current were awarded to AAI due to the efficiency of the system.

Subcontractor Documentation

The EF-111A OFT program consisted of prime contractor AAI as well as two subcontractors. At the outset of the program, AAI recommended the magnetic media/embedded software documentation approach to both subcontractors. One agreed to implement the new approach while the other chose not to.

The subcontractor that chose the new approach made all initial documentation submittals to AAI on magnetic tape. Embedded commands and formats were identical to those used by AAI. Inputs to AAI books were incorporated in a manner similar to any other text file. Subsequent changes to subcontractor documentation were submitted to AAI via red-lined hard copies. This allowed AAI to review all changes prior to incorporation on magnetic media. Advantages to the system were better management control and cost savings.

The subcontractor that did not choose to use the magnetic media approach submitted all documentation to AAI in the traditional hard copy manner. AAI had to subsequently transfer the hard copy to magnetic media either by retyping the material or by using an optical character reader. A burden was placed on AAI to accomplish the transfer, but as a result, better subcontractor control was achieved.

Illustrations

Magnetic media could not accommodate certain illustrations. In these instances, blank pages containing only the figure title and number were inserted in their place. A separate appendix consisting entirely of the illustrations referenced from the blank pages was submitted. This was a hard copy appendix with no magnetic media conversion possible. Figures for all CPPS CPC volumes of a particular subsystem were contained in the same appendix; i.e., Appendix X of the particular subsystem CPPS volume. Therefore, all illustrations for CPPS's were contained in a total of four different appendices, Appendix X in each of the upper level subsystem CPPS's.

TECPD illustrations not presentable on magnetic media were handled in the same manner as CPPS illustrations with one exception. Such illustrations for all TECPD's were contained in Appendix X of the TECPD volume; i.e., each TECPD had its own Appendix X containing illustrations referenced from blank pages within the text.

Special Characters

Some special characters and Greek symbology used in math algorithms were not reproducible on magnetic media. These characters and symbols were replaced with acronyms. An appendix that listed these symbols/special characters and the acronyms that replaced them accompanied each submittal of documents. This appendix appeared in the hard copy versions of the TECPD, Volume I, since it could not be accommodated by magnetic media.

The use of acronyms was a benefit in that all special characters and symbols, which mostly appeared in the CPC CPPS algorithm modeling sections, were fully handled via the system computer. There was no need to manually insert, either via typewriter or by hand, additional characters after the text file was complete. Invariably, when this type of insertion is necessary, some symbols are missed and equations are subsequently incorrect.

Classified Documentation

Tactics was the only EF-111A OFT subsystem containing classified data. Specifically, the classified data appeared in the source code listings, but not in the preambles. These listings went through the same formatting process on the subsystem disc as did all other listings (Figure 2); however, the classified listings contained a character that caused them to go to a special hard copy directory at the end of the formatting process instead of to the publications disc as did the unclassified listings. The classified listings were subsequently printed and delivered as a separate CPPS appendix and were not submitted on magnetic media.

Cost Savings

Various sections of this paper describe in detail the cost saving features of the EF-111A OFT documentation effort. In summary, cost savings for both AAI and the Air Force were realized due to (1) the elimination of multiple hard copy submittals, (2) less document storage space required, (3) the ability to review documents on remote terminals without incurring reproduction costs, (4) the ability to access selected portions; i.e., files of documents without searching through large amounts of hard copy pages, (5) the reuse of disc packs and magnetic media, (6) the ability to copy portions of source files into documents, (7) the elimination of word processing hours, (8) the automatic generation of software family trees, calling trees, input/output tables, etc., (9) the ability to efficiently implement documentation updates resulting from software element changes, and (10) the accurate presentation of source data generated automatically from data base files. AAI also benefitted from incentive fees awarded for concurrency, accuracy, and timeliness of documents. The costs of additional equipment, incorporating nonmagnetic media subcontractor inputs, and generating an additional data item (Magnetic Media Users Guide) were more than offset by the savings listed above. Had this approach been defined at contract award, additional savings would have been realized by all three contractors using the magnetic media approach.

CONCLUSIONS

The results of the magnetic media/embedded documentation approach for the EF-111A OFT were most favorable. The cost savings realized by both the Air Force and AAI merit investigation for possible future implementations of the approach. The most significant aspect of these savings is that they were achieved without compromising the quality or integrity of the product. Further, there was no impact on the program schedule. The Air Force is still benefitting from the approach in that modifications to the trainer, which is now out in the field, are easily and inexpensively documented.

The belief at AAI is that the benefits of the EF-111A magnetic media/embedded documentation approach far outweighed easily resolved problems. In future implementations, areas that warrant investigation are special tools to handle illustrations, special characters, and symbology on magnetic media; expansion of magnetic media use for additional software documents; and increased file manipulation capabilities such as backward scrolling for the user. In addition, it is recommended that the implementation of the magnetic media approach on a program should be a requirement for all participating parties if more than one contractor is involved.

ABOUT THE AUTHOR

Mr. Prince is a Software Publications Engineer at AAI Corporation. He holds a Bachelor of Science in Mathematics and a Masters of Engineering Science. Mr. Prince has 6 years experience in radar systems and flight simulators.

ADA® COMPILER PROJECT MANGAEMENT ISSUES

Wendy J. Hudson
Concurrent Computer Corporation
Tinton Falls, New Jersey 07724

ABSTRACT

The use of the Ada language in a development project impacts the traditional approach to project planning. The experience at Concurrent Computer Corporation in the development of an Ada compiler, written in the Ada language, showed that the design phase of the project was longer than anticipated. The increased design time significantly decreased the system integration time. In addition, the time spent learning to use the Ada language effectively was a large portion of the total project time. The coding rate was not unusual and the overall project schedule was only 10% greater than the original plan. Initial results also indicate that the resulting product is more reliable when written in the Ada language.

INTRODUCTION

In January 1985, Concurrent Computer Corporation began the development on an Ada compiler which was written in Ada. At the conclusion of this project, it became evident that the use of Ada significantly altered the phases of the project.

The results of this project were influenced by project planning and by the background of each of the project team members.

Project

As with most companies, we had not written a large amount of software using Ada. Most of the Ada coding had been done on small and experimental projects. Thus, the Ada compiler project was a new experience for both the management and the project team.

The development of the Ada compiler was split into three pieces. The front-end of the compiler was purchased from Systeam, in Karlsruhe, West Germany. This front-end was selected because it was part of a validated compiler and was itself written in Ada. The front-end of the compiler was to be rehosted to our operating system and upgraded to include some implementation dependent features. The front-end of the compiler contains the syntax and semantic checking. The other pieces, and a major portion of the new code, was the development of the back-end and the run-time system. The back-end contains the code generation and optimization. The run-time system contains the support for things such as I/O, tasking, and operating system support.

The compiler consists of approximately 400,000 lines of Ada code of which 50,000 lines of code were newly generated. The back-end contains about 30,000 lines of code of which 24,000 lines were new. The run-time system contains 21,000 lines of code of which 17,000 lines were new. The modified code in the front-end and other sections was about 9,000 lines.

After integration, the entire compiler would require testing and validation. While this project was similar in size to previous compiler projects, about 60,000 - 80,000 lines of code, the testing of the full 400,000 line system was expected to be much greater than any previous compiler integration. In addition, the Ada validation test suite is much larger and more rigorous than any other compiler validation test suite.

The project was also split across multiple locations. Most of the group were based in New Jersey. There were three people in West Germany working on the compiler front-end changes and another person in Virginia working on code generation.

Project team

The make-up of a project team can often

mean success or failure. In this respect, we were very fortunate.

The size of the group varied during the project. The core group contained 11 people and a full-time manager. For short periods of time, an additional three people also worked on tools and testing. There were four consultants who did specific work in the front-end and the run-time system.

In the core group, 5 of the 11 were very familiar with the Ada language when the project began. Those with experience had been designing and coding in the Ada language for 2-3 years. In addition, they had a very strong background in compiler development. Three others in the group had strong compiler backgrounds, but no Ada programming experience. The remaining three people were newly hired with essentially only academic experience in both compiler development and Ada. All but three people in the core group hold graduate degrees. Generally, the group had a strong academic background with significant compiler design and Ada programming experience.

The manager had worked with most of the group for several years. He had previously managed large compiler projects and had a strong compiler background, but very little experience with the Ada language.

Many of the members of the project team had worked together before on other compiler projects. The group had a good spirit of cooperation, a very professional attitude, and a strong commitment to the project. The group was also quite receptive to the plan to design and code in Ada. They believed that using Ada would benefit the project and they worked very hard to make its use successful.

The experience in Ada programming and compilers along with strong management and team spirit combined to create a very effective programming team. The composition of this team was a critical factor in the success of the project.

Project planning

Before the project was fully staffed and had begun, three senior members of the team spent two months on analysis and planning of the project. They studied the existing code in the compiler and designed the high-level strategy for the back-end and the run-time system. Their research led to a breakdown of the work required and the creation of the initial project plan. This evaluation and definition was one of the key factors in the project's success. It provided the baseline for the project analysis.

For the six members of the group who were unfamiliar with the Ada language, learning Ada became a major activity. We found that Ada is more difficult to learn **and use effectively** than other languages. It is difficult to gauge when someone has gained enough knowledge about a programming language to use it effectively. Generally, this evaluation was done through the design and code walk-throughs. The five experienced Ada programmers on the project made the evaluations easier.

Only one or two people enrolled in a one week introductory course. Most people in the group learned Ada by self-study and read books on the Ada language for the initial instruction. A microcomputer-based course was available which was helpful after the initial introduction to the language.

We had estimated that learning the Ada language would consume about one month of each person's time. After that point, we expected that the individual would be able to code effectively and design small sections of the project. We found that in order to reach this level, it took 2-3 months of training time for each person. As a result, 12 to 18 man-months were added to the schedule.

As a result of our experience, we have developed some Ada "guideline definitions" for people learning Ada. After three weeks of training, a team member was familiar with the Ada language terminology and was considered to be an "Ada coder". After 2-3 months, the programmer became an effective member of the project team. At this point, they were an "Ada programmer". It took nine months to one year before the programmer could effectively design major pieces of Ada code for the project and be considered an "Ada designer". It is interesting to note that the ability to effectively use Ada did not seem related to either the compiler or academic experience of the team member. Experienced compiler designers took as long to learn the Ada language as the new college graduates.

Schedule and manpower results

Table 1 outlines the planned versus actual schedule and manpower results for the design, coding, and system testing phases of the project. The **design phase** was defined as the design of all the interfaces between all the procedures.

Table 1:

Schedule:

	Plan		Actual			
Phase	Duration	% of Project	Duration	% of Project	% Change	
Design	3 months	20%	5.5 months	34%	83% increase	*
Code/Sub-system test	6 months	40%	7.5 months	45%	25% increase	**
System Integration	6 months	40%	3.5 months	21%	42% decrease	
	15 months		16.5 months		10% increase	

Manpower:

	Plan	Actual		
Phase	Duration	Duration	% Change	
Design	28 man-months	53 man-months	89% increase	*
Code/Sub-system test	88 man-months	112 man-months	27% increase	**
System Integration	63 man-months	45 man-months	28% decrease	
	179 man-months	210 man-months	17% increase	

*** - Increase primarily due to additional time for learning Ada**
**** - Increase primarily due to additional functionality**

The **coding phase** included:

- design of the algorithms for each routine
- coding of the routine
- code walk-through
- unit testing of the routine
- testing of the routine in a sub-system

The **system integration phase** included putting the new code generation sections together with the modified front-end, and testing the complete compiler with the new run-time system. The end of system integration occurred when all the Ada compiler validation and internal tests were successfully completed.

The Ada validation test suite checks for conformance to the language standard. At the time that we completed the validation, the suite contained about 1800 tests. The Ada validation tests are more extensive than the validation suites for other languages. However, since it only checks conformance to the standard, we needed to generate additional tests for quality, performance and system dependent features.

Design

People who have embarked on large Ada projects and have subsequently written about them, have all found the design phase to be longer than anticipated. Certainly, our experience in this project was no different. The schedule was 83% longer and the manpower required was 89% greater. In this phase, we designed, coded, and debugged the package specifications to verify the interfaces between all the procedures. The extra time required in this phase was for the debugging process. We feel that the time spent in this debugging effort was the most significant factor in the ease of integration later in the project.

Coding

During the coding phase, the algorithm design, coding, code walk-throughs and testing were done quickly. Table 2 contains information for one piece of the new coding effort for the code generation section of the compiler. This code was part of the critical path in the project. The code size was 38% larger than planned, but notice, the coding took far less time than planned. We feel that the reduced time was a result of the efficiency and features of the Ada language.

Coding was originally estimated at 11 lines per day per person. The estimate was based on the fact that about half the group had never programmed using the Ada language. This also accounted for the procedure design and testing. The coding rate was actually 47 lines per day. If the design and system integration time were included in the calculation, then the coding rate over the entire project was about 15 lines of code per day per person.

It would seem that coding would be shorter than planned, yet Table 1 shows that the coding phase exceeded its schedule. This was due to additional functionality being added to the product. An improved version of the compiler front-end became available. It was decided that we would use the new front-end; it was believed that this could be done within the same scheduled time frame. The new features in the front-end caused a greater number of changes in other sections than was originally anticipated, therefore the coding phase was lengthened.

System integration

The system integration phase was less intense than planned. The schedule was 42% shorter and this phase used 28% less manpower. Within a few days from the start of integration of the compiler, it was compiling complete programs. This is a very unusual situation in the development of compilers. When problems were detected, they were easy to locate and fix. Steady progress was made in locating and resolving all the problems which were found when running the validation test suite.

During the integration phase of previous compiler projects, the number of problem reports generated were usually about 700. 389 problem reports were generated for the Ada compiler project. This number is especially significant since the Ada compiler has 5 times the number of lines of code than any previous compiler project. Also, the Ada language was new to 6 of the team members. We had anticipated that there would be a larger number of bugs as a result of using this new and complex language.

The success of this phase was a result of the early debugging of the design of the interfaces. The use of Ada naturally led to this work being done early in the project. It lengthened the design phase, shortened the integration, and significantly improved quality and reliability.

Maintenance/Reliability

Since validation, the compiler has been used at six customer sites and at numerous Concurrent Computer field office sites as part of the beta testing period. During the test period, only 52 problem reports were generated against the product. Of these 52 reports, only three were compiler problems. The remaining problem reports concerned the packaging and the compiler operation scripts.

Table 2:

Coding Estimates:

Plan		Actual	
3025	lines	4183	lines (38% increase)
275	man-days	89	man-days (68% decrease)
11	lines/day	47	lines/day
87	procedures	87	procedures
35	lines/procedure	48	lines/procedure

Based on these initial results and the few number of problems detected during the integration phase, it appears that the Ada code will be much more reliable than products written using other languages. The reliability and maintainability cannot be accurately measured without more widespread use. Our experience in the integration and beta phases allows us to anticipate exceptional performance in this area.

Other project observations

The development process on this project was much different from projects which used other languages. The emphasis on design, when using Ada, distributes the machine requirements and compiler load more evenly throughout the project. When the Ada specifications are coded and debugged early in the project cycle, the interfaces stabilize early in the coding cycle. After the specifications became stable, the number of recompilations were significantly reduced. Less time was spent by the programmers sitting at terminals to incrementally program the procedures. The use of the Ada language forced the programmers to think more carefully about the programming process.

The recompilation requirements of Ada also changed the approach to modifications. Ada requires that a routine be recompiled when a change is made to another routine that it depends on. Information found in the Ada compiler project library is used to determine these dependencies. The group became careful in changing Ada specifications which would require major recompilations. Such changes were usually saved until the end of the day. The changes were made and the major recompilations could be done overnight. The group felt comfortable with this approach because the recompilations were generally completed without problems. This attention to changes in the specifications also helped focus group attention on critical modifications. Critical modifications were done only after consultation with the rest of the team.

We had originally planned to add more computers to the project as it progressed. Because of the reduced compilation load, this greater computing power was not needed. We did, however, need larger amounts of disk storage than had been planned. The Ada requirement for project libraries and the source control of 400,000 lines of code contributed to the additional disk requirements.

CONCLUSIONS

The Ada language is excellent for programming. Although it requires a greater amount of design time and takes longer to learn, the integration time is substantially reduced and the product is more reliable and maintainable.

Two separate conclusions can be drawn from our experience. The first conclusion relates to new Ada projects using new Ada people:

- expect a 2-3 month period for learning Ada
- expect effective design of major portions of the project only after 1 year of work in Ada
- expect increased design time due to unfamiliarity with Ada

Generally, the members of a project team should begin learning Ada as soon as possible. Greater project time should be allocated to the design phase and whenever possible, the detailed project planning should be done by those familiar with both the Ada language and the application.

The second set of conclusions that can be drawn would be for the project with seasoned Ada designers and programmers:

- expect the design time to increase about 40%
- expect the integration time to decrease about 30-40%
- Over the **entire project** (design, code, integration, test) expect the coding rate to be about 15-20 lines per day per person
- During times of actual **coding activity,** expect the coding rate to be about 40-50 lines per day per person
- expect reduced problems and more reliable software

In closing, the programming and management team are critical to the success of the project. A team that understands and believes in the concepts of Ada will be more effective. A successful Ada project can be accomplished if both the management and project team understand the differences in designing and programming in an Ada system, and expect the advantages of Ada to accrue.

About the author:

Ms. Hudson is Senior Manager of the Scientific and Support Languages development group at Concurrent Computer Corporation. She holds a BA and an MS degree in Computer Science from Rutgers University. She has 11 years of software development experience involving compilers, operating systems and networking. Since joining Concurrent Computer Corporation in 1979, Ms. Hudson has been involved in the development of FORTRAN and Ada compilers, and symbolic debuggers. She is currently managing the group responsible for the development of the Ada, FORTRAN and Pascal compilers.

• Ada is a registered trademark of the U.S. Government (Ada Joint Program Office)

THE JTCG-TSD-ORGANIZATION, INTERSERVICE PRODUCTS AND COOPERATION

Joseph T. Cianfrani
Naval Air Systems Command, APC205-2
Washington, D.C. 20361-1205

Susan P. Cash
Naval Sea Systems Command, PMS-400E
Washington, D.C. 20362-5101

ABSTRACT

In 1976 the Joint Logistics Commanders formed a committee to foster interservice integration of trainer development projects; the Joint Technical Coordinating Group for Simulators and Training Devices (JTCG-STD). The committee's title was subsequently changed in 1986 to the JTCG for Training Systems and Devices (JTCG-TSD) to encompass the scope of complete training, rather than hardware components.

The committee initially met with only limited success but in the last two years it has renewed service enthusiasm. Contrary to current management approaches, the enthusiasm has occurred by imposing an additional layer of oversight into the process. This oversight is accomplished by an 0-6 level steering committee to review its efforts. This additional tier of management, the steering committee, is composed of six members -- the Army's Program Manager for Training Devices, three Air Force members representing the Aeronautical Systems Division, the Logistics Command, and the Human Resources Laboratory. The two Navy members are represented by NAVAIR's Air Program Coordinator for Training and the Naval Training Systems Center's Director.

Because of the right personal chemistry and a commitment for real changes shared by the group and emphasized by its steering committee, the JTCG-TSD has been able to achieve extensive communication and cooperation between the services in the training arena.

The chartered mission of the JTCG-TSD is to maintain technical and management oversight of all activities within the four services which involve joint research and development and acquisition or support of training systems and devices. A project must offer a pay-off to two or more services before it can be sponsored by the JTCG-TSD.

This paper will discuss the management structure of the JTCG-TSD and provide status on its products. For example:

- The Standard DOD Simulator Digital Data Base Common Transformation Project (Project 2851) is to develop a DOD simulator data base and common transformation software, then participate in developing a production facility to make the common land mass data available through the four services to the trainer industry.

- ADA will be specified in all proposals in FY 1987 and beyond by the Air Force and Navy while the Army will require ADA in all mission critical systems. Eight tri-service projects are now underway to make ADA a reality. A detailed summary of these initiatives is provided.

- Finally, a most difficult initiative, Embedded Training, is under close review by the JTCG-TSD. The four services now are addressing possible initiatives to focus talents into areas of mutual concerns (scope and platform types) to encompass a planned embedded training capability.

BACKGROUND

The Joint Technical Coordinating Group for Simulators and Training Devices (JTCG-STD) was chartered in 1976 by the Joint Logistics Commanders (JLC) to encourage interservice integration of research and development projects.

In an attempt to make the group more effective, the JTCG-STD was rechartered in 1982 to identify opportunities to coordinate and consolidate research and technology programs across the entire training spectrum of development, acquisition, operation, and support. It was tasked with maintaining oversight and eliminating duplication of effort, thereby improving economy and efficiency in training device operation. To encompass the scope of complete training, rather than merely the hardware components, the committee's title was also changed in 1986 to the JTCG for Training Systems and Devices (JTCG-TSD).

Between 1982 and 1985, the composition of the JTCG-TSD was not appropriate to identify and take advantage of joint programs. Implementation of the JTCG-TSD plans and studies was a long and difficult process. Thirty-eight groups were reporting directly to the JLC, making their span of control excessive and program delays of up to a year common.

In June 1985, all JLC panels and groups were reviewed by the Joint Secretariat, the Subcommanders Groups, Command Staffs, and the JLC Planners Groups. Because many of the JTCG-TSD tasks had been completed or were well underway, the JLC recommended that the JTCG-TSD be restructured in a staff-to-staff relationship. The JTCG-TSD Chairman seized this opportunity to mold the JTCG-TSD into a highly effective, truly tri-service organization. The Committee analyzed its history and the reasons it had not been effective and found that it had been managed purely within the technical arena. It needed decision makers and managers with broad overviews of each service and its missions. Such managers could commit their services, often on the spot, to the support of interservice projects; so an 0-6 level steering committee was established.

Under the new system, when the JTCG-TSD has an issue significant enough to warrant JLC attention, the committee can work through the Memorandum of Agreement (MOA) signatories. This MOA has been approved at the service two-star level.

The external effect of the restructuring was to foster more independence onto the JTCG-TSD, enabling it to initiate new joint service activities as well as serve as a vehicle to the JLC, if necessary. Along with the name change in 1986, the committee's mission was expanded to encompass the following elements:

-- to maintain oversight of all activities within the four services which involve research and development, acquisition, or support of training systems and devices.

-- to identify and approve specific projects for joint sponsorship which offer pay-off to two or more services by consolidating efforts into a single, joint-sponsored initiative.

-- to ensure coordination and exchange of information between and among the services to minimize or eliminate duplication of effort.

-- to facilitate the exchange of information, such as technical reports and contractor past performance, between and among agencies of the services, to improve the efficiency of operations.

ORGANIZATION

To force coordination between the services, and to ensure a goal orientation, the internal structure of the JTCG-TSD was also revised. The revised structure implemented this expanded scope. The organization now includes three groups of participants (see Appendix 1):

-- An O-6 steering committee composed of the training community's "decision makers" -- program managers with the authority and responsibility for training. These six members include the Army's Program Manager for Training Devices, three Air Force members representing the Air Force Aeronautical Systems Division's Training Systems Program Office, the Air Logistics Command, and the Human Resources Laboratory for Operational Training, and two Navy members -- the Director of the Naval Training Systems Center and the Naval Air Systems Command's Air Program Coordinator for Training. As the service training system focal points with the broad perspectives of their services, they provide overall policy guidance. Since they also have the "purse strings," they can direct their services to fund and support JTCG-TSD sponsored initiatives.

-- Principal members charged with study plan initiation and approval and day to day administrative direction. These members are from the same organizations as the Steering Committee members. They review and approve all studies and interface on topics within their respective services. They also provide consolidated service positions with respect to task applicability.

-- Sub-group chairman and members serving as specific study plan coordinators for visual systems, ADA, etc. They are the individuals actually working the study plans within their service. They are responsible for their products and incorporating them into specific programs. In addition, the Sub-group Chairman must keep abreast of related interservice tasks and make all information available within the Sub-group.

The Steering and principle committee membership has been kept small to avoid the indecision characteristic of large committees. As specific issues arise, the chairman invites appropriate members to sit in on the proceedings and present their organizations' views. Figure 1 shows the organizational structure.

JTCG-TSD ORGANIZATION

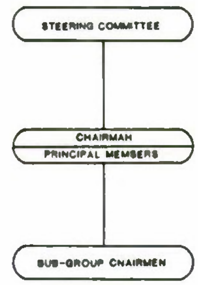


FIGURE 1

PROCEDURES

Under the new organization, the process for study plan initiation, evaluation, approval and lead service appointment has been greatly streamlined (see Figure 2). The committee tries to keep management simple. There are, accordingly, no formal steps in program initiation. A tri-service program proposer can simply contact any principal member who then acts as a committee filter/advocate. After discussion with the member, the proposal concepts will then be presented for review by the six principal members at the next Quarterly Meeting. If the proposal is deemed to have multiple service application of significant magnitude, a draft study plan will be requested by the Chairman. The study plan is prepared by the program initiator. At the following Quarterly Meeting, the principal members will review the draft plan and approve or disapprove it. If the plan is approved, the study will be briefed to the Steering Committee at the bi-annual meeting. The Steering Committee will appoint a lead service, which will appoint a subgroup chairman. Emerging issues, such as the need for additional service funding or assigned personnel requirements, will also be addressed.

To be accepted by the JTCG-TSD, a study plan must relate to joint service training, be funded by at least the lead service, and be product oriented. In addition to identifying products for specific applications, the JTCG-TSD conducts quarterly intergroup reviews and an annual briefing to the Steering Committee. The reviews provide an opportunity to get high level interservice visibility for the products.

STUDY PLAN PROCESS

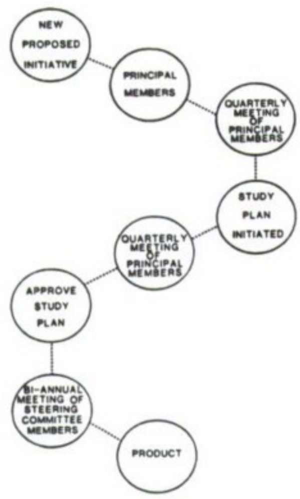


FIGURE 2

Concern for the overwhelming number of study plans which could be initiated led to the JTCG-TSD's decision to process the study plans by combining them under task related subgroups. Hence there are only six JTCG-TSD approved study plans, but each plan contains two to eight related tasks or products. Appendix 2 contains a complete listing of these study plans and tasks.

Combining numerous tasks under one study plan has a two-fold advantage. One, it forces intercommunication between detailed technical levels and makes interservice products more readily available to a larger number of related technical areas. Two, the Steering Committee's review of the tasks provides the broad range visibility throughout the services needed to achieve more varied applications of the products or enhance them with additional support.

JTCG-TSD STUDY AREAS

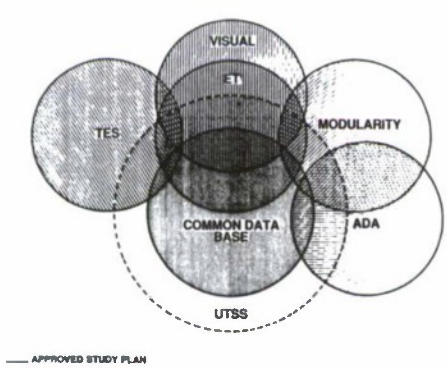


FIGURE 3

STUDY PLANS

The following discussion will present a description of some of the more interesting tasks currently underway under the JTCG-TSD study plans. Limited descriptions of all current JTCG-TSD Committee tasks, focal points, and schedules are found in Table 1. A fuller description can be found under the same item number in Appendix 2, while Figure 3 shows the interrelationship between study plans. Most of the tasks selected for discussion in the body of the report have large service/industry interest with results and products expected before July 1988.

IMMEDIATE PAYOFF TASKS

All studies have conducted, at minimum, limited cost trade-off analyses. For the purposes of this paper, the savings will be documented as intangible. However, if warranted, details of the analyses may be obtained from sub-group chairmen.

A. Modular Simulators

The first area to be discussed is Modular Simulators (Table 1, Item B). Task 2 of this study plan covers the Multiple Micro-Computer System (MMCS)/Advanced Development Model and Semi-Automatic Partitioning Tools and Assessments. This task has been undertaken by the Naval Training Systems Center. Four problem areas are under investigation: control of microprocessors, data bus contention, software partitioning, and cost. A micro-computer test bed utilizing software from NTSC's VTRS (SEL 3275) was developed and attached to a low cost A-4 cockpit. The feasibility of using aggregate processors and their efficiency was examined. The final report has been completed, showing that partitioning can be accomplished and that the semi-automated partitioning concept was feasible. In addition, the report contains recommendations on efficient utilization of DOD-STD-2167 for multi-processor systems.

A follow-on effort has been proposed by NTSC to assess the use of multiple micro-computers to real-time system computational tasks: specifically compatibility with MIL-STD-1553B data busses, applications to aero software models and relationship between total computing power and number of micro processors.

B. ADA Insertion Program

A second area which promises early payback is the ADA Insertion Program (Table 1, Item C). Tasks include the Air Force's Cost Model Survey and the ADA Simulator Validation Program (ASVP). The Cost Model Survey was a six month effort to evaluate cost estimating models for applicability to ADA software acquisitions. The Survey utilized two parallel ASD/ENETC small business initiatives to evaluate current cost models for ADA reusable software to support ADA acquisitions. The two contractors evaluated JENSON, COCOMO, RCA PRICE and other cost models and published separate and unique concepts. One contractor developed a new designating system which it called "Archetypes," to simplify the estimating model, using an EXPERT system. The second company developed a system methodology to control the basic system multipliers and automated methodology to incorporate collected data from industry.

The second task area, ASVP, is also using two contractor teams (Burtek/Intermetrics/CCC and Boeing/SAIC/Gould CSD) to redevelop real-time ADA software packages for the C-141B operational flight trainer and the E-3A full flight simulator respectively. Both contractors are using different design methodologies to broaden the knowledge base.

Burtek's critical design review (CDR) occurred 27 May 1987, and the Air Force demonstration will take place in December 1987. Boeing's CDR and demonstration, using redeveloped ADA software will be completed by November 1987. A lessons learned session with industry is planned for late in FY 1987.

C. Visual

The last area of immediate payoff is in the visual category (Table 1, Item 3). Task 1 consists of a tri-service test program to evaluate Head/Head-Eye coupled displays. A test plan has been written and will be used to evaluate PM-Trade's/AFHRL's Advanced Visual Technology System (AVTS) display (which utilizes fiber optics as the image input), AFHRL's FOHMD (Fiber Optic Helmet Mounted Display), and NTSC's helmet-mounted scanning laser/dome display system Visual Display Research Tool (VDRT). Engineering performance test data for all five of the visual tasks will become available in June 1988; however, the data from the three above systems will be gathered by October 1987. Thus far, standard test procedures and a common test pattern specification have been written. A common test fixture (a gimble mounted head fixture with photometer, laser pointer and other test jigs) has been developed.

ADDITIONAL TASKS

Standard DOD Simulator Digital Data Base/
Common Transformation Program

An important JTCG-TSD task with a mid range payoff is the Standard DOD Simulator Digital Data Base/Common Transformation Program (Project 2851).

The recurring costs associated with data base development within the simulator industry, along with the possible lack of future data base availability, have induced the Air Force Aeronautical Systems Division (ASD) to attempt a tri-service standardization program.

The program will develop software producta for use in simulator training devices requiring digital cartographic data. It will use a common land mass data base containing Defense Mapping Agency data and transforming mechanisms. ASD will develop the software. First deliverables are scheduled for 1991.

All preliminary work (assessing technology and available sources, and defining requirements) has been completed. Conaequently, in May 1987, ASD awarded a contract for the first phase of development of a demonstration prototype system to a PRC/GE/Hughes team.

When the outputs are validated, the Navy will utilize provisions in its F-14D contract to incorporate the new data bases. The services will decide on a full production facility location in the latter part of FY 1987.

NEW STUDIES

The JTCG-TSD is currently focused on three additional areas: a Tactical Environmental System (TES), embedded training, and universal threat simulation systems (UTSS). TES and embedded training are JTCG-TSD approved studies. UTSS is currently under investigation and discuasion. In the latter area, the JTCG-TSD is in the process of identifying the common denominators between the services and defining the final products. A study plan has been requested for this initiative.

A. Tactical Environment System (TES)

Under TES, the Air Force will develop and test a system to evaluate pilot performance and create training profiles. The Navy will link multiple F-14D trainers located at a single training site into a common mass attack tactical training problem. This will include five cockpits and 160 active targets with 12 interactive targets.

B. Embedded Training (ET)

More tri-service work needs to be done on a coordinated basis on embedded training. Although service requirements vary greatly, the JTCG-TSD has agreed to provide complimentary products to each service's task. Fourteen areas of ET will be investigated. As psrt of the overall planning, the Steering Committee directed the JTCG-TSD to

1. Develop a common tri-service definition of embedded training.

2. Obtain OASD approval for the tri-service definition.

3. Evaluate the effectiveness of currently in-place ET systems in the areas of air, sub and surface. Determine problems/constraints to implementing ET and analyze for R&D or administrative tasks.

4. Develop a central ET data base and conduct a case study of pending acquisition.

The tri-service definition of embedded training was tentatively accepted by OASD and will be a diacussion topic at the ninth annual I/ITSC. The JTCG-TSD helped considerably in cutting through red tape within the individual services and getting the definition to OASD. An embedded training study plan is now being drafted with the Army as the lead service.

C. Universal Threat Simulation Systems (UTSS)

A draft study on universal threat simulation . now being developed by the JTCG-TSD. The JTCG-TSD recognizes that this area is a high cost driver and expects an approved study to result in early 1988.

CONCLUSION

The JTCG-TSD is an active organization trying to achieve cost efficiency. Data transmissions and aggressive management are the key to a more efficient utilization of our research and development talents within DOD. In this paper we have tried to show some immediate payback tasks that the JTCG-TSD is currently engaged in and how the committee

operates. Appendix 2 provides details of the studies as extracted from the approved study plans. As new tri-service initiatives occur, the JTCG-TSD will maintain its open-mindedness and attack problem areas that provide useful products, concentrating more on short term payoffs than extended studies. Direct and intangible cost savings are available through a wide variety of tasks underway. However, it is each manager's responsibility to select and expand on those areas which apply to his own programs. Although the JTCG-TSD is available to provide information, managers must visualize where the services are heading and take initiative in asking for the information. Without active involvement by each manager, the work of the JTCG-TSD will not realize its full potential.

ABOUT THE AUTHORS

Mr. Joseph T. Cianfrani - currently chairman of the JTCG-TSD - is employed by NAVAIRSYSCOM, APC205, as Deputy for Aircrew Systems Training. The position is responsible for engineering, training and logistics on all Navy aircrew trainers. Previous experience includes positions as Professor of Acquisition Management at the Defense Systems Management College, Chief Engineer of the Navy's 5" guided projectile and Program Manager for the 5" MK 45 naval gun system. He holds a Professional Engineer's License, an M.S. in Systems Engineering, a B.S. in Mechanical Engineering, and is currently working on an M.S. in Systems Engineering.

Susan P. Cash is a Program Planning Specialist for the Research, Development, Test and Evaluation portion of the Aegis Shipbuilding Program, Naval Sea Systems Command. She was previously employed as a Logistic Management Intern in the Aviation Training and Logistics Management divisions of the Naval Air Systems Command. Ms. Cash holds an M.A. in American Government and Politics from Georgetown University and a B.A. in the same area from Centre College. She is proficient in four languages and studied French Literature and Civilization at Montpellier, France.

APPENDIX 1

JTCG-TSD Committee

MOA Signitaries

ARMY	-	M. L. Ferguson BG, U.S.A.M.C.
NAVY	-	R. C. Gentz RADM, NAVAIR
AIR FORCE	-	R. S. Steer MG, AFSC
	-	A. T. FARRINGTON BG, AFLC

Steering Committee

ARMY	-	R. Lunsford COL, USA PM-TRADE
NAVY	-	J. Vomastic CAPT, USN NAVAIR APC205
	-	C. Rowley CAPT, USN NTSC/00
AIR FORCE	-	W. Lobbestael COL, USAF ASD/YW
	-	B. Pearson, II COL, USAF ALC/MMF
	-	M. Lane COL, USAF AFHRL/OT

Principle Members

ARMY	-	Dr. R. Hofer PM-TRADE/AMCPM -THD-E (305) 646-4747 AV 791-4747
NAVY	-	Mr. J. Cianfrani * APC205-2 (202) 692-2137 AV 222-2137
	-	Mr. F. Summer NTSC/741 (305) 646-4437 AV 791-4437
AIR FORCE	-	LTCOL E. Clayton ** GQ AFSC/SDTA (301) 981-3487 AV 858-3487
	-	Mr. W. Lloyd HQ ASD/YWB (513) 255-7177 AV 785-7177
	-	Mr. B. Bishnow HQ AFLC LOC/TGW (513) 257-7060 AV 787-7060

* Chairman 86/87
** Chairman 85/86

Study Plan Chairman

See Appendix 2

APPENDIX 2

Common Data Base

Standard DOD Simulator Digital Data Base/
Common Transformation Program
(Project 2851)

I. SCOPE. Project 2851 will serve all DOD simulator training devices requiring the use of topographic data. First, DOD will develop a standard simulator data base (SSDB). The SSDB will be founded on error-corrected Defense Mapping Agency (DMA) source data, enhancement and additions to DMA source data, and libraries of models and texture patterns. The SSDB will be updated and augmented by a data base generation/modification capability that will incorporate, maintain, and configuration manage enhancements to DMA source data. Second, DOD will develop common transition software that will convert the SSDB into component generic transformed data base (GTDB) formats with sufficient flexibility to be used for future visual and sensor simulations. Third, DOD will manage the large amount of data within SSDB and GTDB and allow timely access to information regarding availability of data and its level of enhancement. Fourth, DOD will design and develop a flexible software system, using modern software engineering techniques and the ADA language to permit a wide variety of modifications for future growth.

II. TASKS

A. Data Base and Software Development

This is a contracted effort to develop and demonstrate a prototype system. The common transformation programs will be developed with sufficient flexibility to adapt to changing technologies and not inhibit the inherent advantages of competitive innovation. All software will be developed in ADA using modern software engineering techniques, ensuring sound program design and allowing for ease of expansion and modification of the software throughout its life cycle. The effort is scheduled for a 30 month period commencing June 1987.

B. Interim Production and Validation

The data base and transformation programs will be evaluated on operational image generating systems. The contractor will demonstrate Project 2851 interim operation by generating, enhancing, transforming, and maintaining an additional area of standard data base in support of a low risk simulator program. Interim production will include improvements to the software generated during the development phase.

The contractor will develop an implementation plan for an operational facility to support all services. This phase is scheduled to be an 18 month effort commencing upon completion of the development phase.

C. Production Facility Identification and Deployment

A data production facility is to be established. However, an exact definition of the facility resources will not be available until well into the development phase.

Modularity

Modular Simulators

I. SCOPE. A basic requirement of the modular simulator design concept is the capacity to add, delete or change major subsystems or modules without impacting other subsystems/modules. Thus, no longer will one simulator vendor develop the vast majority of the training system. Instead, it will be possible for many small companies to design and develop standard modules that may be usable/reusable on several simulator projects. Moreover, the standardization of modules will reduce trainer support costs on each project that implements the modular design approach.

II. TASKS

A. U.S. Air Force

The Modular Simulator Design program consists of three phases -- Request for Information (RFI), Modular Concept Definition Study, and Proof of Concept. The first two phases have been completed.

The third phase will be a multi-year effort starting in FY 1987. This effort consists of design definition and a demonstration validation. The final implementation approach and architecture will be a consolidation of the design studies, DOD guidance and inputs from the simulator industrial community.

B. U.S. Navy

The functionally Modular Multiple Micro-Computer System (MMCS) is an advanced development of a model distributed microcomputer system that can be expanded to perform a wide range of trainer applications. Distributed processors use the Eurocard circuit boards and communicate through the high speed VME bus. In the first phase of this effort, now completed, trainer software was partitioned into functionally modular sub-routines and executed by modular hardware (Eurocard circuit board) VME modules. A semi-automated software partitioning approach was developed to reduce partitioning errors and to expedite software development. A real-time flight simulation was the first demonstration on the ADM. Four potential problem areas for a closely coupled distributed system were investigated: (1) control of the micro-computers; (2) a bus contention; (3) software partitioning; and (4) software development and update costs. The issues of standard interconnect definition and development of generic control algorithms were also addressed. A follow-on effort will perform the following additional tasks:

1. Assess the compatibility of MIL-STD-1553 busses with inherently distributed multiple micro-computer architectures.

2. Apply avionics, aero and threats software models.

3. Establish the relationship between total computing power and a number of micro-computers in a training system environment.

C. U.S. Army

The simulation complexity test bed is a variable fidelity helicopter simulation system intended

to provide a low cost test bed capability for empirically determining the minimum fidelity and other simulation characteristics required for effective rotary wing training. The initial capability will consist of major stand-alone components of attack helicopter simulation capable of man-in-loop experimental manipulation and subsequent integration with other major components in building block fashion.

The Aviation Combined Team Trainer (ACTT) contract award is scheduled for the fourth quarter of FY 1987. The program provides for the development of scout modules (OH-58 aircraft and AH-1P), attack modules (AH-1S and AH-64), a battle captain module (scout configuration), and instructor station. Each of these modules will be capable of interaction.

ADA

ADA Insertion

I. SCOPE. An ADA Insertion Program, to better understand system software design methodologies and impacts on software acquisition, has been established to collect acquisition and support data for future aircrew training devices implemented in ADA.

II. TASKS

A. U.S. Air Force

1. The Deputy for Training Systems ADA Simulator Validation Program (ASVP) task involves redevelopment of the real-time software packages for multiple training devices using the ADA language. Burtek and Boeing Companies are to examine the technical and management issues of ADA implementation in simulators.

a. Burtek has teamed with Intermetrics and Concurrent Computer Corporation (CCC) to redevelop the C-141B Operational Flight Trainer. They are using a refined object-oriented design methodology, augmented with structured analysis and an ADA-based PDL. Burtek has scheduled a two-year effort culminating with a demonstration of the redeveloped software.

b. Boeing has teamed with Science Applications International Corporation and Gould CSD to redevelop the ADA software for the E-3A Full Flight Simulator. Boeing is using a top-down object abstraction methodology augmented by structured analysis utilizing a tiered design, code and test approach rather than using a PDL.

2. ESD/AC awarded two six month Small Business Innovative Research contracts in July 1986 to survey existing software cost estimating models for applicability to ADA software acquisitions.

3. Cost of ADA reusable software. The Deputy for Training Systems is initiating a six month effort beginning in January 1987 to evaluate the cost impacts of software reusability concepts in both the short and long-term timeframes. Also, approaches to obtaining and managing reusable software will be examined by this effort. Contractor: TASC, Boston, Massachusetts.

B. U.S. Navy

NTSC's ADA insertion program consists of the following four tasks:

1. Experiment with using ADA on a training device (an F-4 Aircraft Weapon Systems Trainer which had been previously coded in FORTRAN). The following are some of the results:

a. No perceptible difference in flying qualities.

b. ADA took twice as much processing time as FORTRAN (used up to 50% spare time).

c. Memory required was the same or less than FORTRAN.

2. ADA Risk Assessment Contract. An engineering investigation support task, entitled "ADA Risk Assessment" was issued to the University of Central Florida on 31 August 1985. This task generated ADA benchmarks representative of three current complex aircrew trainers. Time and memory comparisons with the original FORTRAN were made for these functions.

3. Equipment Operator Trainer (Device 14E36X). The plan is to procure one device and use it as an acoustic target generation tool in the Research Department at the NAVTRASYSCEN. The device is to be built around multiple Motorola 68010 microprocessors.

C. U.S. Army

PM-Trade's ADA Insertion program consists of a UH-1FS ADA Systems Engineering Feasibility Project (Task 5617). Research into the use of ADA on flight simulators will be conducted using the UH-1 Flight Simulator as the research vehicle. The trainer software, which is currently written in assembly language, will be redesigned and implemented in ADA. The trainer computer system will be replaced with a modular design which will allow the four-cockpit configuration to be separated into two two-cockpit trainers at a future time. Metrics on programmer productivity and other aspects of the ADA design process will be collected.

Visual

Test and Evaluation of Head and Head/Eye Coupled Visual Simulation System

I. SCOPE. Test and evaluation effort of head and head/eye coupled visual systems currently under development by the three services.

The prime objective of this program is to characterize, in a common context, the various head and head/eye coupled visual systems being developed by the three services. Common test parameters and procedures have been developed. Engineering performance tests and utility applications will be conducted on five head and head/eye coupled visual display systems.

II. TASKS. The Air Force Aerospace Medical Research Laboratory is developing a binocular head-coupled helmet mounted display with CRT image input. The imagery is currently generated by a calligraphic computer image generator (CIG). The Air

Force Human Resources Laboratory (AFHRL) is developing a head/eye coupled display based on the aame baaic helmet display optics, but with fiber optics as the image input. The Navy Training Systems Center and the Air Force Training SPO are jointly developing a head/eye coupled projector/dome visual system. The Army is developing (under a AFHRL contract) similar head/eye coupled projector/dome visual systems. While several of these visual systems are similar in overall concept, the implementationa are quite different and unique. Each development project has test and evaluation requirements. However, the requirements are not common, making comparison of the different head and head/eye coupled visual systems difficult and inconclusive.

In this effort, developed a set of test parameters common to the various head and head/eye directed visual systems such as resolution, field-of-view, brightness, contrast, and response times. The parameters measured were common to all the visual systems. Testing techniques were made as uniform as possible. Utility evaluation and analysis will begin in June 1987. Engineering performance tests will be conducted on the following systems, in the order listed:

1. Navy (NTSC) Visual Display Research Tool (VDRT).

2. Army/Air Force (PM-Trade and AFHRL) Advanced Visual Technology System (AVTS).

3. Air Force (AFHRL) Fiber Optic Helmet Mounted Display (FOHMD).

4. Air Force/Navy (NTSC, Training SPO and Singer IR&D) Eye-Slaved Raster Inset Program (ESPRIT).

5. Air Force (AFAMRL) Visually Coupled Airborne System Simulator (VCASS).

In addition to the engineering performance tests, a display utility evaluation will be conducted. The resulting documentation will be in the form of a series of reports, one covering each system, and a summary report. The analysis will involve projected acquisition and operational cost, aa well as engineering performance, concentrating on the capability and utility of each visual system.

Tactical Environmental System

Advanced Combat Simulation for Tactical Aircraft

I. SCOPE. The Air Force (Deputy for Training Syatems) will conduct an indepth study of training requirements baaed upon current threat assessments for future Air Force tactical training. The training requirements will be validated on prototype hardware leading to the definition of a training system in the form of training objectives and hardware needed for the system. The Navy at this time is procuring hardware to meet the tactical environment training objectives of the F-14D Maritime Air Superiority (MAS) mission. The Naval Air Systems Command is procuring as a part of the F-14D simulator program the capability to link multiple F-14D trainers at a single training site into a common tactical problem to train tasks such as fleet defense against mass attacks.

II. TASKS

1. The objective of the Air Force Advanced Tactical Combat Simulation (ATCS) is to define the training requirements that exist between today's current training system capabilities and the training requirements posed by current threat assessments. A comprehensive list of tactical combat training objectives will be defined for fighter aircraft. Each objective will include the conditions, performances and performance standards required for task accomplishment.

After the development of the objectives and corresponding conditions, an evaluation plan of the different tools/technologies available to assess, measure and quantify each standard will be prepared.

A baseline will be developed to support the user in his generation of a Statement of Need.

2. APC205's Tactical Environment System (TES) purpose is to link five F-14D mission trainers into one mass attack problem involving 160 active targets and 11 interactive targets.

The Preliminary Design task will identify the TES design criteria, math models, interfaces, preliminary data base and other information necessary to establish the development configuration. The Detail Design task will complete the detail design solutions.

After the detail design is completed, the trainer hardware will be fabricated, software coded and integrated at the aubaystem/configuration item level and tested. Detail design documentation will be updated as necessary to form the basis of the product specifications.

Embedded Training

Embedded Training (ET)

I. SCOPE. The scope of the study plan is to analyze and consolidate the coordinated tri-service initiatives, to establish a systematic ET methodology and associated data base specifying embedded training requirements in weapon systems acquisition efforts. The methodology will enhance the training requirements definition process and optimize the total training concept through trade-offs of such issues as life-cycle cost, reliability and maintainability, safety, and training effectiveness.

II. TASKS

1. Evaluate effectiveness of in-place embedded training systems (air/surface/submarine) including maintenance trainers.

2. Investigate fourteen selected tri-service classes or subclasses of materiel items to determine problems, constraints and impediments to ET implementation. Analyze resulting data to define potential research programs or administrative action that will develop solutions to those problems.

3. Develop/centralize embedded training data base.

4. Conduct study of SV-22 for potential embedded training applications.

TABLE 1

1987 JTCG-TSD SPONSORED STUDY PLANS

SUBJECT (Program Title)	Sub-Group Chairman/#/Service	Task Titles/Description	87 Funding	Program Description	Product Availability
A. Common Data Base (Standard DOD Simulation Digital Data Base/Common Transformation (Project 2851))	Major M. Sieverding A/V 785-7177 USAF		$ 3.0M	Generic Transformed Data Bases from DMA Data	
		1. Development and Demo Prototype System			1/90
		2. Interim Production and Validation			6/91
		3. Production of Services Required for Data Bases			6/91
B. Modularity (Modular Simulators)	CAPT J. Coates A/V 785-7177 USAF		$ 3.2M	Standardize and Transport Software Modules	
		1. Proof of Concept (Validated MIL-STD)			10/91
		2. MMS/Advanced Development Model and Semi-Automatic Partitioning Tools aod Assessments			12/86
		3. MMS Follow-on/Evaluation and Application			TBD
		4. Simulation Complexity Test Bed			7/93
		5. Modular Trainer/AH-1S and AH-64 Team Trainer			4/93
C. ADA (ADA Insertion Program)	Mr. B. Lloyd A/V 785-7177		$ 3.9M	Effective Implementation of the ADA Language	
		1. ADA Simulator Validation Program (ASVP)/Redevelopment of Software Packages on C-141B/E-3A			11/87
		2. ADA for Project 2851			6/91
		3. Acquisition Model Survey			1/87
		4. Reusable Software Cost Survey			7/87

SUBJECT (Program Title)	Sub-Group Chairman/#/Service	Taak Titlea/Deacription	87 Funding	Program Description	Product Availability
		5. Evaluation of ADA Design and and Reusability Methods			2/88
		6. Equipment Operator Trainer (Device 14E36X)/Test Bed			3/89
		7. Evaluation of Hardware/Software Design Characteristics/ADA Code for F-18			10/87
		8. UH-IFS ADA System Engineering Feasibility/Assembly Language Redesigned and Implemented in ADA			10/89
D. Visual (Teat and Evaluation of Head and Head-Eye Coupled Viaual Simulator Systems)	Mr. R. Ewart A/V 785-2431		$ .16M	Evaluation of Head/Head-Eye Visual Systems	
		1. Develop Parameter and Test Procedures			9/87
		2. Evaluate AVTS			9/87
		3. Evaluate VDRT			9/87
		4. Evaluate VSCDP/FOHMD			10/87
		5. Evaluate Eaprit			11/87
		6. Evaluate VCASS			12/87
		7. Final Analysis and Documentation			8/88
E. Tactical Environment Simulations (Advanced Combat Simulation for Tactical Aircraft)	Mr. J. Keller A/V 222-1966		$ 2.5M	Develop Pilot Evaluation Criteria and Link Flight Simulators into Common Tactical Problems	
		1. Advanced Combat Tactical Simulation			1/94
		2. Tactical Environmental System			8/92
F. Embedded Training	Mr. D. Peckham A/V 791-5881	Develop Common ET Capability	$ 0	Evaluation of 14 Systems	12/88
G. Universal Threat	Mr. S. Gibb A/V 222-0947	Develop Common Threat Data Baae	$ 0	TBD	TBD

STONE AGE TRAINING IN A SPACE AGE ENVIRONMENT

Lt Colonel Rowland H. Worrell, III
1013th Combat Crew Training Squadron
Air Force Space Command
Peterson AFB, Colorado 80914-5000

ABSTRACT

Air Force Space Command was established in September 1982 to conduct operational missions in space. The need to support those missions with well-trained personnel led to the creation of Undergraduate Space Training, an organization tasked with providing its graduates with a broad base of space fundamentals, and the 1013th Combat Crew Training Squadron, a unit which provides system specific operational crew training. The courses provided by both schools were designed using Instructional System Development technology and utilize a media mix which includes lecture, computer based training systems and simulation. This paper addresses the problems of developing training programs and acquiring simulation capability to support training personnel stationed at more than 30 sites worldwide with missions that vary from flying satellites to warning of missile attack. The paper also discusses the use of networked desk-top computers to provide space operations center simulation and explores the management decisions required to determine proper media mix. It compares training results of the previous on-the-job training programs with new, full fidelity simulation. The paper closes with comments concerning training programs and simulation as an integral part of new space system acquisitions.

INTRODUCTION

During the birth and evolution of Air Force Space Command (AFSPACECOM), various missions were drawn together from agencies throughout the Air Force. The training programs supporting these missions, however, were routinely nonstandardized and depended heavily upon on-the-job training (OJT). While the Department of Defense has long used OJT methods for upgrade training, the lack of standardization and ineffecicencies inherent in OJT programs become training shortfalls when OJT is used for initial qualification training (IQT). These deficiencies, added to the risk of having students training with on-line equipment used to operate critical national systems, have characterized AFSPACECOM training--an inferior system which remained stagnant as operational requirements increased in number, duration, and technical complexity. In other words, AFSPACECOM had been using a stone age training system in a space age operations environment.

Any training system, however deficient, could profit from a systematic scrub of requirements and formalization of instructor lesson plans and other course control documents. For several years the Air Force has employed an excellent course development process called Instructional Systems Development (ISD) which provides the tools for repairing defective courses and developing new formal training. The bigger problem, however, lies not in revising classroom presentations, but rather in getting training off the operational equipment. The solution to that problem is simulation.

But the acquisition of full fidelity simulation equipment often presents severe technical and managerial challenges because of the wide variety of missions within space operations, the uniqueness of the many operating systems, and the small number of students that train for each system annually. A computer based training system (CBTS) may be the answer to some of these challenges.

THE NEED

If a trainee crashes an aircraft, the unit may have lost one vehicle in a fleet of 200. But if a trainee sends a bad command which disables a satellite in a single vehicle constellation, the unit and the nation may have lost the whole fleet. The danger of having a trainee passing information over a command and control network upon which our national leaders depend cannot be overestimated. However, operational risk is not the sole rationale for establishing a formal, off-line training program.

The space operations career field has grown significantly over the past several years in assigned missions and number of personnel. From its embryonic size of 500 operators in 1981, the field will expand to over 1900 by 1990. This rapid growth will inevitably be accompanied by a corresponding decrease in operator experience levels. A solid training program offers the only defense against this shrinking experience base and provides the only means to offset the impact of rapid personnel changes.

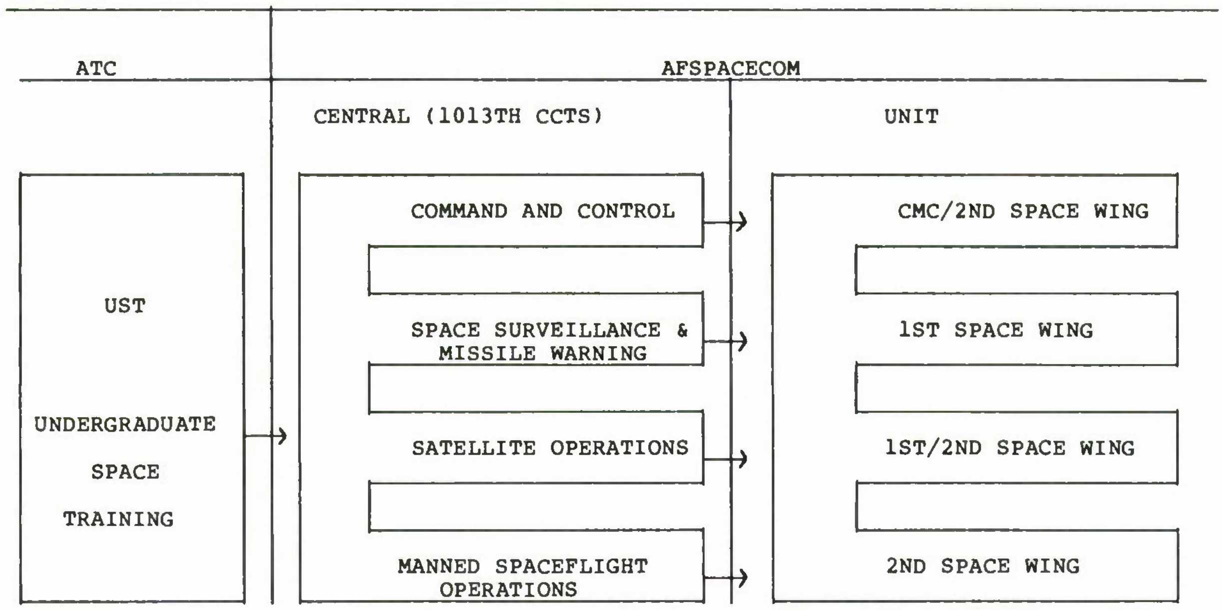


Figure 1. Concept for Space Operations Training

A TRAINING PLAN

The space operations career field includes four distinct areas: early warning, command and control, satellite operations, and manned spaceflight. In 1984 no existing organization or training system could capably handle the diverse set of training requirements attending these specialties. To correct the deficiency, AFSPACECOM and Air Training Command (ATC) conducted a joint review of space operations training requirements and established a training plan (Figure 1) which would not only meet current needs but also support the growth of the space operations career field. The plan called for a clear division of training responsibilities between ATC and AFSPACECOM.

Undergraduate Space Training (UST)

ATC assumed responsibility for developing a course of instruction which would provide a broad base of space knowledge to students preparing for a career in space operations, much the same as Undergraduate Pilot Training provides broad based, hands-on experience to pilot candidates. UST graduates could then enter any of the space operations specialties.

As developed, the school includes academics, computer based training (CBT), and generic simulation. The simulation not only teaches the skills needed for console operation but also emphasizes stress management to ensure students possess the characteristics required to work effectively in space operations centers. The simulation is generic in nature and does not require upgrade every time an operational system is upgraded. However, the operations skills trained and scenarios presented closely parallel the operational environment.

Crew Positional Training

AFSPACECOM's role in this new training concept is to provide crew positional training through a Combat Crew Training Squadron (CCTS). Following graduation from UST, new space operators come to the CCTS for system specific training. This training qualifies them to operate a crew position in a space operations center. The CCTS also provides a centralized schoolhouse to which space operators return for retraining prior to reassignment to new space systems or to upgrade to instructor status.

CCTS graduates are mission capable: they know the system, its checklists, malfunctions, and events but are not authorized to operate console positions unassisted by a certified operator. When graduates report to their unit of final assignment, they are assigned to operational crews. There they learn local site procedures and crew integration. Then, with their operational crew, they take the final check, qualify mission ready, and become certified operators authorized to work unassisted.

Like UST the CCTS uses academics, CBT and simulation in its training programs. Unlike UST the training is mission and position specific. CCTS instructors, therefore, must be fully

qualified in the systems they teach, and all simulation must provide a large measure of fidelity with the actual equipment on which the students will later qualify. Additionally, whereas the basics which UST teaches rarely need updating, operational system changes and upgrades create a problem of currency for CCTS lesson plans, CBTS software, and simulation software and equipment. This challenge will be discussed as we visit each instructional medium individually.

The plan shown in Figure 1 is still in its infancy, but already there are significant reductions in on-site training times, and operators exhibit better system knowledge. The continued growth and success of this plan depend largely on quality courseware development and the continued acquisition and enlightened use of CBTS and simulation capabilities.

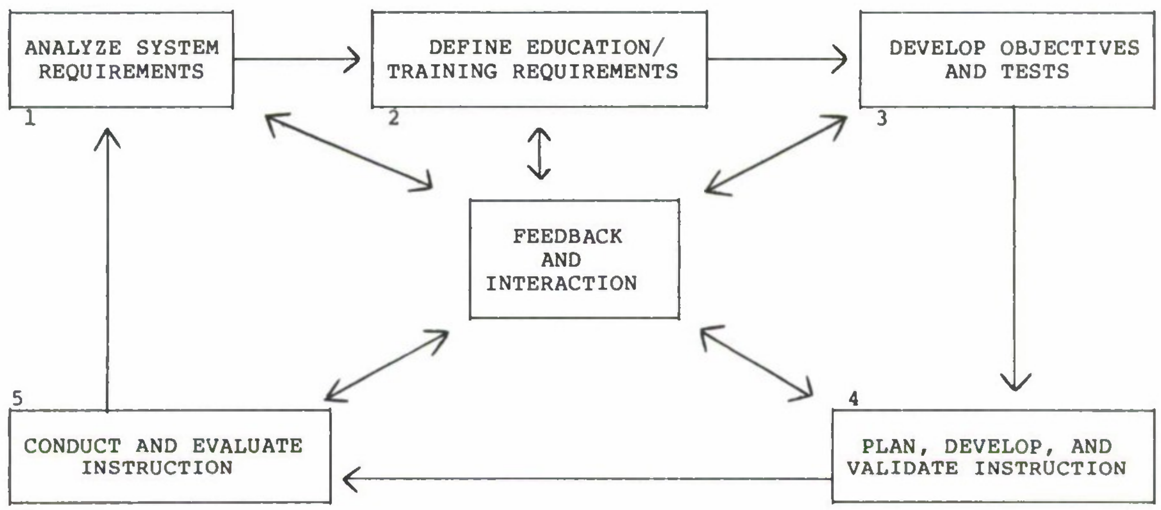


Figure 2. Air Force ISD Process

COURSEWARE DEVELOPMENT

Courseware development is a structured process which identifies training requirements and corresponding instructional methods. If the process is conducted properly and judicious course management decisions are made, then the product of these labors is a workable, cost-effective instructional system.

The Air Force subscribes to a process called Instructional Systems Development (ISD). It is nothing more than good management applied to training and the application of a systems approach to development and execution. All training components are logically interrelated. Each component has its own function, and each has an effect on other components. A change in academics affects the simulator, and adding new simulation equipment affects academics. The entire training system is an integrated whole.

The ISD process involves the five steps shown in Figure 2. The key to this process is that each step produces a predictable, quantifiable set of products. Each step utilizes the products of the previous step. The result is a chain of documentation into which changes can be readily infused.

Basically ISD uses actual job data from the field to formulate objectives and determine what needs to be trained. It then designs student centered courses which teach and test the objectives. Thus, student progress may be precisely measured against specific standards. This process, when combined with accurate, reliable feedback, results in an effective and efficient course of instruction.

MEDIA

The ISD process fails unless it accurately identifies the correct media by which each objective should be trained and reinforced. Classroom lecture, CBT and simulation are the primary tools used in instruction.

Classroom Lecture

With few exceptions, classroom lecture has been the backbone of Air Force academic instruction. The instructor is typically a subject matter expert and more often than not has operational experience in the system being taught. Assuming the lesson plans have been prepared in accordance with the ISD process, the lectures will be well structured, accurate and effective.

The advantages of the lecture medium include the ability of the instructor to enliven and personalize the class with his/her own operational experiences. It also allows the class to ask questions and receive an immediate response. As procedures change or new events take place, course material can be updated almost instantaneously. One instructor can teach a group of students simultaneously and typically requires no equipment other than a slide projector, overhead projector and chalkboard. Most importantly, the lecture medium provides an authentic, credible presence to motivate the class. In the space operator business, where assignments can be to the far extremes of the globe, motivation plays a significant role in how well students learn and subsequently how well they perform.

The few disadvantages of lecture include lack of opportunity for individualized instruction. Lecture is typically too slow for some and too fast for others. In addition, the instructor's ability and the attitude he conveys may be less than positive and present a potential liability to the class. Thus, ensuring quality control in the classroom becomes a more difficult task. Finally students are typically in the receive only mode and rarely afforded the chance to actively participate in the teaching process.

Full Fidelity Simulation

Conversely, full fidelity simulation is almost totally interactive. It puts students into a realistic semblance of the environment in which they will work. It usually pairs them one-on-one or two-on-one with an instructor. Lessons are precisely structured and the simulation is carefully controlled to ensure all students get the same information.

Despite the procedural diversity of space operations mission areas, all operators share the commonality of working from computer driven consoles which present radar or numerical data. Thus simulation is fairly easy to devise. By acquiring actual site consoles and a driver which will produce all site displays, targets and malfunctions, simulation becomes exact. Full fidelity simulation then provides the perfect environment in which to train and evaluate students. The obvious drawback, however, is that space operations consoles and computers to drive the simulation are expensive. System upgrades often require additional equipment be added to the system, thereby increasing the cost. Since many space operations sites are unique and replace only 10-15 operators per year, it becomes difficult to justify such an expense for each system. For those systems with low annual IQT requirements, a more cost effective method of training is required.

Computer Based Training Systems

CBT puts a student at a desk-top computer working self-paced through courseware displayed on the screen. Using an interactive process, the computer software will lead the student through the material, retraining wherever the student displays a lack of understanding. Instructors remain available to answer student questions. Recent studies indicate CBT is more efficient than the lecture method, and students consistently demonstrate better retention.

Using sophisticated software graphics or video disks, operator consoles can be simulated on the CBTS screen with an impressive degree of visual fidelity. Graphically displayed console switches can be "operated" by light pen or keyboard, producing true-to-life equipment reactions. Additionally, CBTS consoles can be networked to interact and provide full ops crew integration. CBTS simulation may not provide the realistic "feel" of sitting at a full size operations console, but it has the capability to simulate many different systems by simply changing the software. Obviously, the cost savings of buying software instead of consoles and computers is tremendous.

CBT, however, is not without its shortfalls. CBT is still a relatively new field, and courseware development and simulation software design tend to be labor intensive. The industry statistics for courseware production show 200 to 500 development hours for every hour of courseware produced. Simulation software is even more labor intensive. Also, authoring languages still tend to be non-user friendly. In many systems extensive programming

skills are required to make even simple program changes. In view of the propensity for change that current space operations systems display, it is imperative that the CBTS be reprogrammable by a line instructor with only minimal programming skills.

THE MANAGEMENT CHALLENGE

Within the ISD process the first real management decision comes in determining the appropriate mix of lecture, CBT and simulation. Additionally, a management decision must be made concerning full fidelity simulation versus CBTS simulation. Both decisions must look at three factors: effectiveness, efficiency and cost.

For training to be considered effective, the student must reach specified levels of proficiency. Training effectiveness increases as students attain consistently higher proficiency levels. Efficiency of training deals with the amount of time required to attain a given proficiency level. Cost deals with dollars spent to acquire and maintain the training program. It includes both equipment and instructor expenses.

Together these factors define the overall goal of improved training productivity--better trained people in less time for less money.

As previously discussed, advanced technology--CBT and simulation--should increase efficiency and effectiveness. However, CBT systems and simulators require capital investments which cannot be ignored. Thus, a cost/benefit analysis must become a part of any course design effort and focus upon life cycle costs, changeability of the operations system, and the ease with which training media can respond to those changes. The decision process must evaluate whether the payback from increased efficiency will be greater than the cost of making it efficient.

NEW SYSTEM ACQUISITION

Each new system acquisition includes a training program for the initial cadre of operators. And although the Air Force pays for that training, rarely is it adequate for a continuing training program. To offset that deficiency and ensure new space systems acquisitions are delivered with a quality training program, a training office has been established within AFSPACECOM's Directorate of Plans. To assist them, the CCTS has developed a standard for courseware development, and a standard for CBT use is being coauthored by AFSPACECOM, ATC, and Air Force Systems Command.

RESULTS TO DATE

AFSPACECOM's CCTS started classes in January 1986 and graduated 682 students from seventeen different courses in its first year. Although many of these courses are still in the development/validation phase of ISD, the results are promising. Courses which were developed using lecture only have resulted in an average reduction of 35% in on-site training time. Courses utilizing lecture and CBT have experienced up to 45% reduction, and courses which include simulation have resulted in reductions of over 50% in unit training time. The training courses which incorporate full fidelity simulation have shown better results than those using CBTS for simulation, but only on the order of 10 to 15%.

Note, however, that these efficiencies have not been without corresponding increases in costs. While each hour of lecture required an average of only 31.7 hours of development time, each hour of CBT courseware required 292 hours and each hour of CBT simulation took 480 hours to develop. Finally, while constructing system tapes for full fidelity simulators took only 10-15 hours, the cost of equipment was prohibitively high at $1 to $5 million per simulator.

CONCLUSIONS

Although the space operator training courses are still young, several conclusions can be drawn.

An OJT program is too inflexible, unstandardized, and dangerous to be used for initial qualification training. It just cannot keep pace with a career field as technically complex and rapidly expanding as space operations.

Although any formalized academic program will shorten training time in the field, the greatest benefits are realized from a program which includes simulation that exposes students to the operational environment in which they will work.

Despite the fact that a live instructor tends to motivate better than a desk-top computer and can rapidly and inexpensively modify lesson plans, well-designed CBT courseware provides student paced efficiencies which classroom lecture cannot duplicate. However, courseware development typically requires computer programming skills and is still a labor intensive process. Full fidelity simulation yields the best training results. But for those systems with a low annual student load, CBTS simulation may be a more cost effective alternative.

Each operational system must be carefully reviewed to identify training requirements. Management must then analyze the need and predict course life cycle costs as a decision factor in acquiring an economical training system.

The challenge to industry is to develop user friendly CBTS authoring languages which can be used by both programmers and instructors.

As DoD budgets tighten, training is historically one of the first places to feel the pinch. But subjugating training is much like the Fram Filter Man saying, "Pay me now or pay me later." Only by smart, aggressive management can the Air Force acquire the cost effective training systems which will build a mature space operations training program...a program that will not only meet the demands of today, but also provide well-trained men and and women to meet the challenges of tomorrow.

ABOUT THE AUTHOR

Lieutenant Colonel Worrell is the Commander of Air Force Space Command's 1013th Combat Crew Training Squadron. He activated the squadron in December 1985 and has directed its growth from 12 to over 120 instructors and support personnel. He holds a Bachelors Degree from the Air Force Academy in Astronautical Engineering and a Masters Degree from the University of Colorado in Aerospace Engineering. He qualified as a Shuttle Landing Supervisory Officer and worked Shuttle missions as part of NASA's Flight Control team. In addition Lt Col Worrell spent fourteen years instructing and evaluating in numerous Air Force aircraft systems including the F-111D.

SHUTTLE MISSION TRAINING FACILITY UPGRADE

Kurt Frevert and Dr. Riley McCafferty
The Singer Company — Link Flight Simulation Division
Houston, TX

ABSTRACT

The Shuttle Mission Training Facility (SMTF) consists of three Shuttle Mission Simulators (SMS) and several lesser training devices. The SMTF Upgrade program is required to improve the capability of the SMTF to train shuttle flight crews and mission support personnel without impacting current capabilities. In partial satisfaction of this requirement, the SMTF Upgrade Step 1 program will replace and upgrade the three SMS computer complexes and rehost the existing software to the new computers using off-the-shelf products where possible. Fundamental to the Upgrade Step 1 task is the need to maintain the existing training capability until the computer upgrade is fully proven. This paper first describes the present SMTF facility and then discusses the hardware and software upgrade concepts and implementation plan to show how current training capabilities are preserved during development.

INTRODUCTION

The SMTF is one of NASA's major mission support elements. It is the principal simulation device used to train astronauts and ground support personnel in the operations of the shuttle vehicle and cargo/payload systems. It provides simulator training of the entire shuttle mission from prelaunch through launch and orbit, including placement and retrieval of various experimental payloads and subsequent descent and landing. Additional preparatory benefits include the validation of shuttle flight software and Mission Control Center (MCC) operational hardware and software via integrated simulations.

The objective of the SMTF Upgrade program is to improve the capability of the SMTF to train shuttle flight crew and mission support personnel without impacting the ongoing operations. In order to accomplish this objective, the SMTF Upgrade Step 1 will:

- Replace the existing computers
- Rehost existing software functions to the new computers
- Use off-the-shelf products where practical

Each SMS consists of up to 30 computer processing units, which respond to and drive the controls and displays in the simulated shuttle flight deck and associated instructor and operator stations. Upgrading the SMS computers in this environment offers some significant challenges. The existing Univac 1100/44 host computer on each SMS will be replaced with a Sperry 1100/92 computer. The existing Perkin-Elmer 8/32 and 3250 minicomputers will be replaced by a single Concurrent Multi-Processor System (MPS) (3280MPS) computer on each SMS. Each computer facility must be modified as necessary while retaining the capability to support classified activities and training. In the software area, the Sperry Operating System (OS) on the 1100/92 computers and the Concurrent OS on the 3280MPS computers must be modified to support real-time simulation. These modifications must maintain existing software structures to continue to support simulator interfaces.

It is essential to maintain the existing training capability until the upgrade SMTF is fully proven. A control and monitor isolation and switching system was developed to place the new equipment and software into operation for debug and test before the old equipment is removed. This allows the old training systems to be used intact during normal training periods and new training equipment to be tested with the training base.

CURRENT SYSTEM

The SMTF simulators are located in two buildings at the Johnson Space Center (JSC) in Houston. The simulators in JSC Building 5 are depicted in Figure 1. The Fixed Base (FB) and Motion Base (MB) Simulators include a Shuttle Vehicle Simulator (SVS) and a Payload Simulator (PLS). The Network Simulation System (NSS) and the Spacelab Simulator (SLS) have stand-alone capability, or can integrate with either the FB or the MB Simulators. The term "Shuttle Vehicle Simulator" is used to denote the core of the Fixed Base, Motion Base, and Guidance and Navigation System (GNS). That core includes the Univac host, the Crew Instructor/Operator Station (CIOS) Intelligent Controller (IC), the Visual (VIS) IC, the Simulation Interface Device (SID) IC, and the Input/Output (I/O) IC. Each set of four IC's is referred to in this paper as the SVS IC's. Figure 2 depicts the host and SVS IC's as part of the SVS.

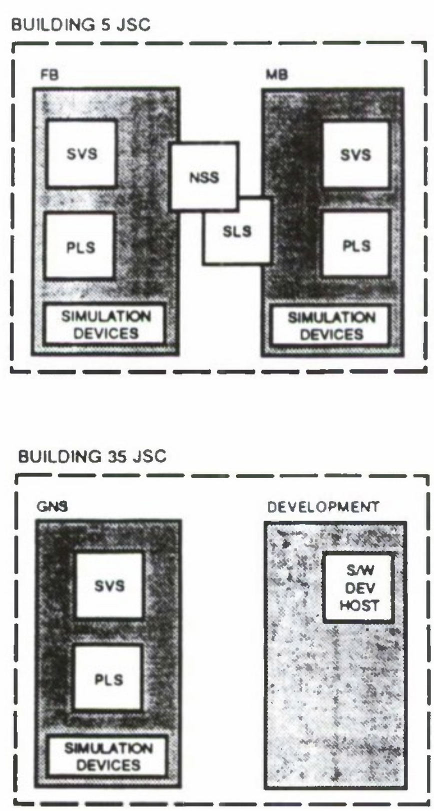


FIGURE 1 SMTF SIMULATOR AND DEVELOPMENT SYSTEMS

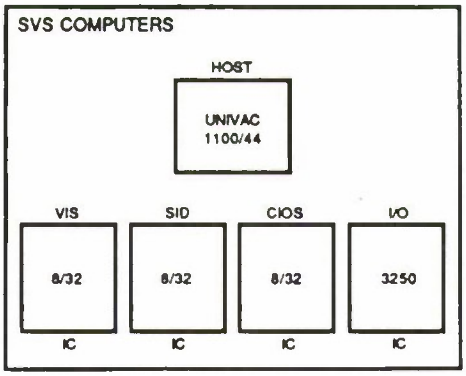


FIGURE 2 SVS HOST AND IC'S

The JSC Building 5 fixed-base and motion-base simulations as presently configured are shown in more detail in Figure 3. The MB and FB hosts are switchable between bases by means of the system select/isolation and configuration switch. These switches are controlled by the Control and Monitoring Isolation System (CMIS). The CMIS consists of two identical microprocessor-based computer systems, identified as the master and standby systems. Each microcomputer system consists of a microcomputer (DEC PDP-11/23), a terminal for operator input and configuration display, a line printer for logging, and interface units to the isolation switches. The operating system and application software of each system are the same. The master system performs both control and monitor functions, while the standby acts as a redundant monitoring system, in effect double-checking the master monitor function. The IC sets are base-specific and operate on a shared memory bus. Each base is made up of a crew station, an instructor station, a SID housing five general-purpose computers and a multipurpose CRT display system, and a visual image generator and display units. The described base configurations are defined as fixed base or motion base.

The SMTF systems in JSC Building 35 at JSC are depicted in Figure 1. The GNS includes an SVS and a PLS. The development systems include the Sperry development host with supporting peripherals. The computers that comprise the SVS in the GNS include one Univac 1100/44 host computer and a set of four IC's. These of four IC's are identified in name and function as the SVS IC's and are as described above for JSC Building 5. The development host consists of a Sperry 1100/91.

Figure 4 depicts the configuration of the GNS simulator and development system. The GNS simulator is made up of the same simulator computer components as described for the FB and MB simulators. Note that the GNS does not have the visual system, which makes up the FB and MB simulators. The development system is used for load build and non-real-time development of software.

The SMTF software is a collection of FORTRAN, assembly, and microcode routines that simulate the real-world elements of the shuttle vehicle/payload system and provide an operating environment for the development and real-time execution of the shuttle vehicle/payload system models. The applications modeling and support software is distributed across the various computer systems within the facility.

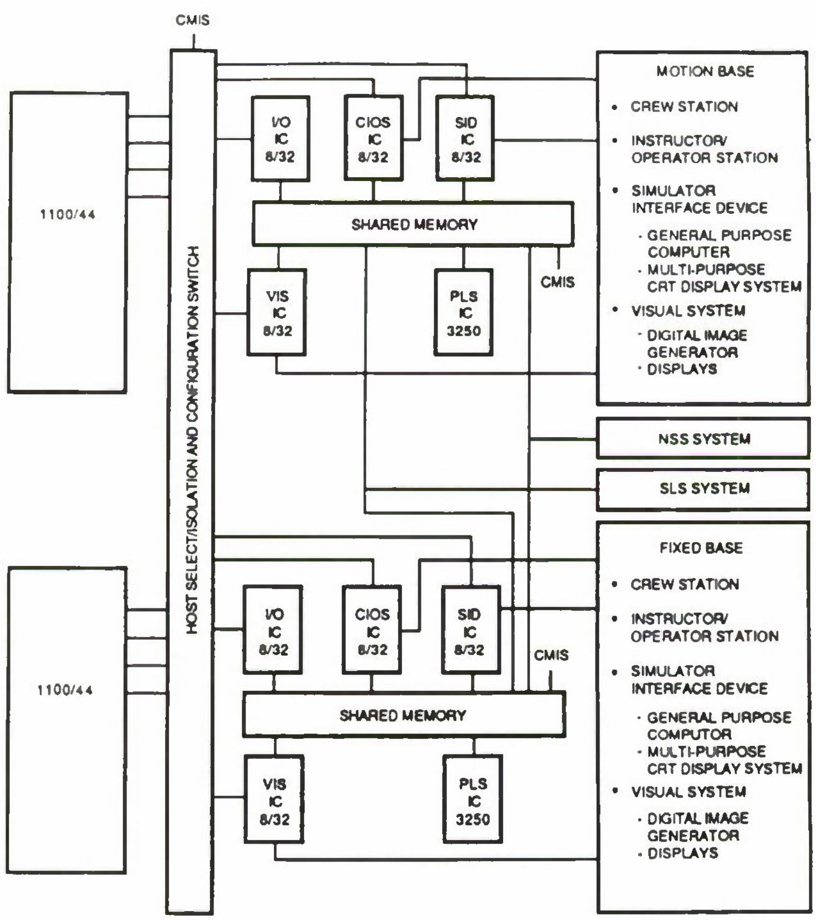


FIGURE 3 JSC BUILDING 5 PRESENT CONFIGURATION

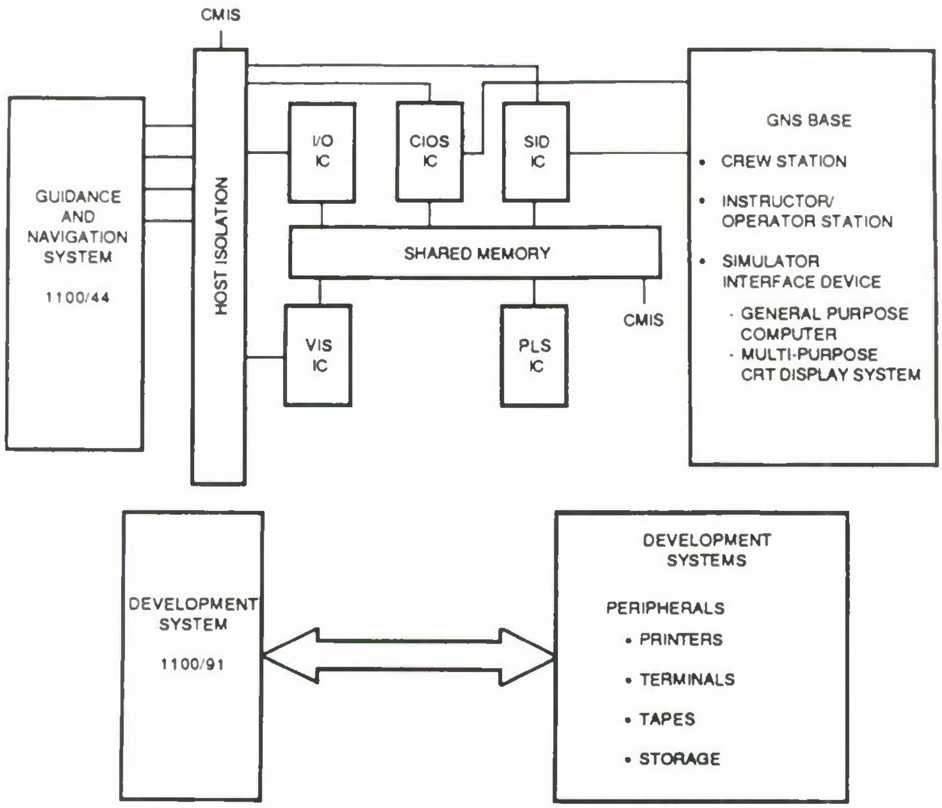


FIGURE 4 JSC BUILDING 35 PRESENT CONFIGURATION

The SVS host computer contains real-time support software that controls the execution and operation of all resident simulation software (both applications and other support software) and handles communications with the SVS IC's. The SVS host applications software models the vast majority of the shuttle onboard vehicle systems Communications, Tracking, and Instrumentation System, the Electrical Power System, the Environmental Control and Life Support System, the Mechanical and Hydraulic Power System, the Propulsion System, the Payload Support System, the Guidance, Navigation and Control System, and part of the Data Processing System, and provides the shuttle vehicle dynamics environment simulation.

Each of the SVS IC's contains real-time support software that manages all software within the computer, communicates with external peripheral simulator unique devices attached to the IC, and interfaces with the host for data transfers. The CIOS IC houses cathode ray tube text and graphics display software for the SVS IOS and software to control the hardware linkage for crew station controls and displays. The SID IC contains Data Processing System modeling applications software that processes data exchanged between the host shuttle onboard systems models and the shuttle onboard general-purpose computers via a Simulation Interface Device. Visual processing code to drive the SMS visual simulation system and shuttle Data Processing System telemetry software (including code to link to a secondary SID I/O port) are located in the VIS IC. The IOIC holds interface software that allows the SVS to communicate with other simulator systems within the SMTF (NSS, SLS, PLS).

The PLS IC contains payload simulation software that interfaces heavily with the SVS applications models. The SLS provides software that simulates all functions of a specialized payload, Spacelab. Shuttle uplink/downlink, tracking, and communications network station simulation code resides in the NSS.

The VIS Digital Image Generator System contains digital image processing and display software to dynamically generate, in conjunction with visual hardware, the out-the-window visual scenes within the crew station.

In addition, several computer systems are utilized in an off-line mode to provide software development features for real-time systems described above. Source code change (edits/compiles/links) and load build software plus configuration management and some flight-to-flight reconfiguration product generation software are resident in the Sperry 1191 development computer system.

UPGRADE CONCEPT

Hardware

The purpose of the SMTF Upgrade is to upgrade the computer complexes for the FB simulator, the MB simulator, and the GNS. This upgrade will be accomplished by replacing the existing Univac host computers and the existing Perkin-Elmer IC computers.

The two Sperry computer systems (1100/92) replace the two existing Univac 1100/44 host mainframes in the JSC Building 5 facility. The FB and MB simulators shown in Figure 5 make up the JSC Building 5 facility. The CIOS IC, VIS IC, SID IC, IOIC, and IC core shared memory on each base were replaced with a Concurrent Computer Corp. 3280MPS. Each host connects to a base IC via four word interface channels. The existing base simulator peripheral hardware is retained, to the extent possible, and interfaced to the new computer systems. The PLS, NSS, SLS, and DIG interface to each base IC's memory system. Isolation switches between the hosts and the IC's are retained. The PLS, NSS, SLS, and DIG's also remain isolable from the base

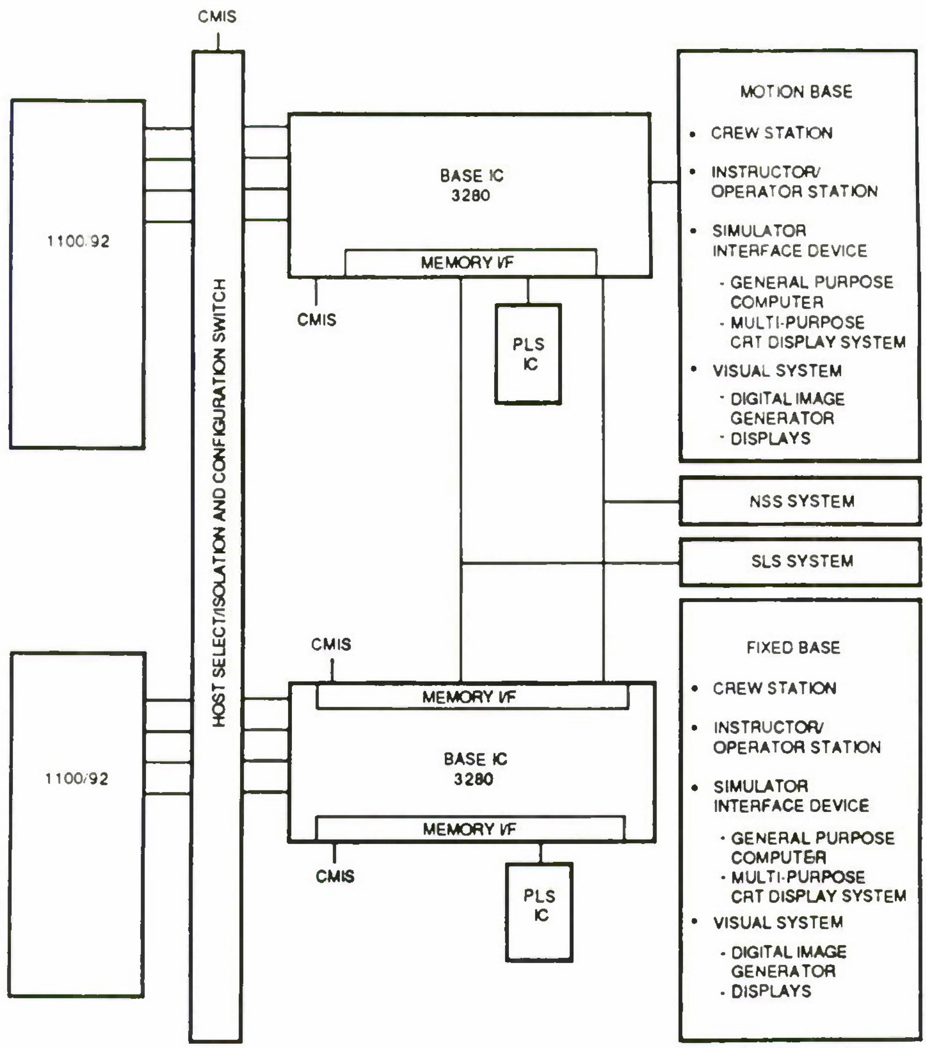


FIGURE 5 JSC BUILDING 5 UPGRADED CONFIGURATION

IC's. Base IC isolation switches will be added to provide switching of training bases and computer systems during interim configurations. All isolation is controlled and monitored by the Control and Monitoring Isolation Subsystem (CMIS), which includes all switching necessary for systems configuration and control.

A high-level system diagram for JSC Building 35 is shown in Figure 6. The upgrade to the JSC Building 35 computer complex consists of upgrading the current machine (Sperry 1100/91) to a Sperry 1100/92. The Sperry 1100/92 replaces the current machine and interface to the GNS 3280MPS via isolation switches and performs functions now performed on the CIOS IC, SID IC, IOIC, and VIS IC. The Sperry 1100/91 development system will not be replaced as part of SMTF Upgrade, but its functions will be retained by also utilizing the GNS as a development host.

<u>Software</u>

This section describes the software modifications required to support the upgraded hardware components in the SVS. There are four major divisions of software to be discussed: operating systems, real-time support software, off-line support software, and applications software.

<u>Operating Systems.</u> Two operating systems will be required for the SMTF upgrade task: Concurrent OS/32 for the CC-3280MPS and the Sperry OS for the 1100/92. These operating systems will replace the real-time monitor used on the existing SVS IC's and the Sperry OS revision used on the existing SVS host.

1) *Sperry Operating System* — A current release of the Sperry OS is used on the Sperry 1100/92 host computers. Local modifications must be made to the OS in order to support real-time simulation. The local modifications applied to the OS used on the 1100/44 computers are the basis for the local modifications needed on the 1100/92 computers.

2) *Concurrent Operating System* — The latest revision of OS/32, which is capable of supporting a CC-3280MPS with Auxiliary Power Units (APU's) and O/P processors, is used for the CC-3280MPS base IC. Local OS code modifications are implemented in order to provide real-time simulation support functions.

<u>Real-Time Support Software.</u> The real-time support software for the SMTF upgrade task is divided into three categories:

1) *Moding and Control* — The master control software in the host is modified to account for the reduction in the number of IC's from four to one. That portion of the safe store logic that is dependent on the current frame job structure of the host load is modified to recognize the new host load structure. The existing master control programs for the CIOS IC, SID IC, VIS IC, and IOIC are consolidated and redundant code eliminated. Modifications to operating system interfaces are made to the base IC master control program to allow it to run under OS/32.

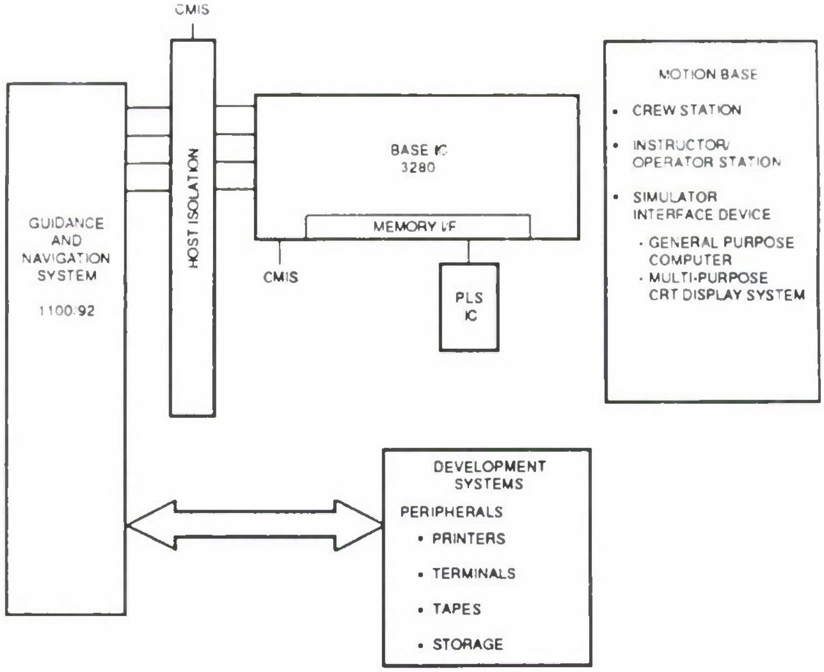


FIGURE 6 JSC BUILDING 35 UPGRADE CONFIGURATION

The host-resident real-time I/O software is modified to optimize the host/IC data transfers. With the reduction in the number of IC's, data transfer blocks can be eliminated or merged to minimize the time required to transfer data in real time. Real-time I/O software for the CC-3280MPS base IC was developed. Real-time I/O subtasks were established to make the most efficient use of processors capable of doing data conversions (CPU or APU). Data conversion routines, capable of running on a CC-3280MPS, were developed.

Operating system interfaces in the support software for the on-board computer software system were modified to run under OS/32.

2)*Sequencing and Control* — Simulation sequencing in the upgraded host and base IC is accomplished by defining a set of tasks which can operate on a queue of jumplists until the queue has been emptied. The structure of jumplists and jumplist tables on the CC-3280MPS base IC is based on program dependencies and execution frequencies. Sequencer code, common to all processors, is used to process the IC jumplists. Execution control on the base IC is accomplished through the creation of a subtask control subsystem. The subtask control subsystem also provides fault handling for intercepting and handling subtask execution faults without disabling the entire simulation. Real-time macro libraries for use on the CC-3280MPS base IC are developed to provide an interface to the real-time operating system.

Figure 7 depicts a block diagram of host sequencing and control. Sperry 1100/92 host tasks and their associated jumplists are structured based on program dependencies, execution frequencies, and transport lag dependencies. Each task contains sequencer code for processing host jumplists. The priority of each task will be defined to the host operating system, which is responsible for task dispatching. Task priorities are established so that transport lag processing is assigned the highest application priority, followed by high-rate non-transport lag and finally lower-rate tasks. A set of interrupt routines is assigned the highest priority with the simulation.

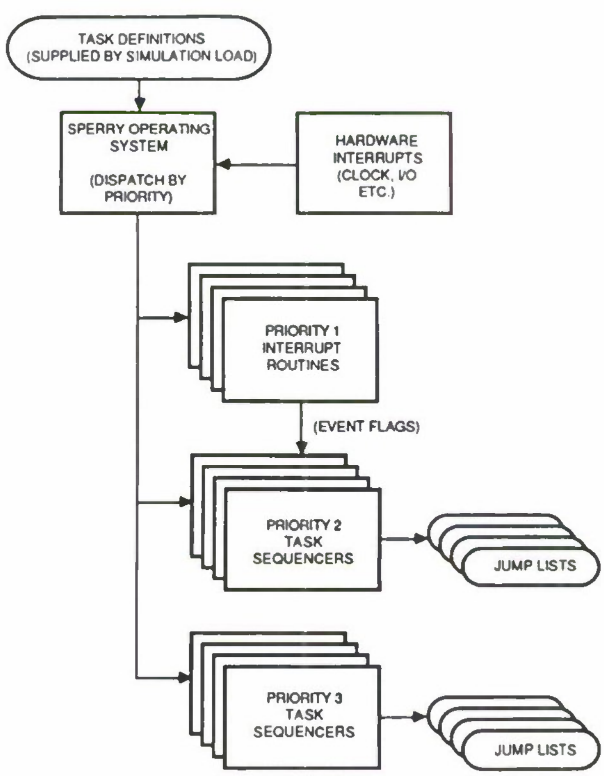


FIGURE 7 HOST SEQUENCING AND CONTROL BLOCK DIAGRAM

3)*Data Display* — The consolidation of the SVS IC's into the base IC provides for simplification of the data display subsystems. The Aydin support software currently residing in the SID IC, VIS IC, and IOIC will be eliminated. The Aydin software residing in the CIOS IC is modified for use in the base IC. These modifications account for the change in the number of IC's and the interface required for OS/32.

The host-resident SVS data logging and data retrieval subsystems are modified to support the change in the number of IC's. Data logging and data retrieval software residing in one of the existing SVS IC's is used in the new base IC. Software for these functions, along with their associated host/IC transfer buffers currently residing in the other three SVS IC's, is eliminated.

Off-Line Support Software. The off-line software for the SMTF upgrade task consists of three major categories: linkage software, data generation software, and control and diagnostic software. In each case the software is modified to support a smaller number of IC's in the simulation while maintaining the SVS interface to the PLS, NSS, SLS, and Digital Image Generator.

Applications Software. Applications software is relinked to the restructured data pool and rehosted to the new computer systems. Host-resident applications modules that depend on the existing host frame job structure are identified and modified to operate in the new host environment. IC-resident applications modules that make calls to the real-time monitor are identified and modified to run under OS/32.

IMPLEMENTATION PLAN

Implementation Overview

The implementation of the SMTF Upgrade program is divided into two major steps. The initial step is performed by upgrading the GNS and development systems in JSC Building 35 and the second step by upgrading the two simulators in JSC Building 5. This implementation plan allows for proof of concept of basic hardware and software development in the Building 35 development facility before starting the JSC Building 5 training facility upgrade.

Hardware Implementation Plan

Interim configurations for JSC Buildings 5 and 35 are required to complete the upgrade implementation task with minimal impact on the ongoing operations. The interim configurations provide for the switching of the current and upgraded computer systems to a training base, so that ongoing operations can continue with minimal impact during upgrade development. This system of switches is under control of CMIS operations, which permits fast and error-free reconfiguration from computer system to training base.

The CMIS features operation support functions such as menus, prompts, status displays, and audible alarms to detect violations and failures. With the CMIS operational control of the isolation switches, the systems may be reconfigured and validated by an operator in a matter of seconds. All transactions and violations are automatically logged and time-tagged to provide configuration records.

By using CMIS to control the interim configurations, crew training may be conducted from 8 to 12 hours a day on a training base. The upgrade development may use the same training base during the remaining hours in the day. This is based on minimum time for reconfiguration of the training system and high probability of being able to return to a training configuration by the next day.

As depicted in Figure 8, the 1100/91 development machine is upgraded to an 1100/92 and a Concurrent 3280 is installed and interfaced to the 1100/92. An IC select switch is installed to select the current computer set or the upgrade configuration for GNS base use. Shared memory interface and switching are installed to permit switching of the PLS.

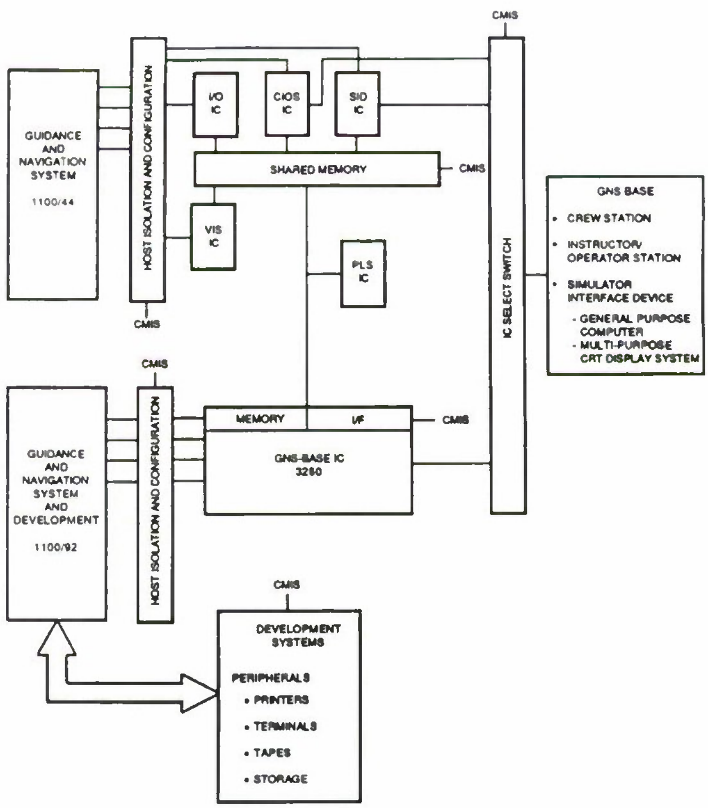


FIGURE 8 JSC BUILDING 35 UPGRADE INTERIM CONFIGURATION

The installation of the IC select switch and shared memory interface to the 3280MPS provides for the continued operations of JSC Building 35 as well as the proof of concept of hardware required for the JSC Building 5 upgrade effort.

JSC Building 35 remains in the interim configuration and continues to support JSC Building 5 operations until Building 5 enters its final upgrade configuration. Figure 6 depicts the final upgrade configuration for JSC Building 35. Note that the 1100/44, 8/32 IC's, and IC select switch have been removed.

Upon the completion of proof of concept in JSC Building 35, the initial steps will be taken to start upgrading of the JSC Building 5 training complex. The JSC Building 5 interim configuration has two phases. Phase 1 is defined by Figure 9 and Phase 2 by Figure 10.

The following steps have been undertaken to install Phase 1:

- Install the 3280MPS base IC and interface its memory to FB and MB shared memories, FB and MB PLS computers, NSS, SLS, and DIG visual system
- Install 1100/92 and interface to 3280MPS base IC
- Install and interface with base and IC select switch

The Phase 1 hardware configuration is required so that the upgraded configuration may be interfaced with either training base. This allows for the upgraded system to be fully operational for the fixed and motion training bases prior to removal of the first 1100/44, 8/82 computer set.

The hardware configuration to support JSC Building 5 Phase 2 is obtained by the following major steps:

- Remove an 1100/44 and MB IC's
- Install an 1100/92 and MB 3280MPS base IC
- Recable the host selection and isolation switch and the base and IC select switch

This Phase 2 configuration allows for old computer (1100/44, 8/32) loads to continue to be utilized for training and does not require them to be rehosted to the upgraded computer systems (1100/92, 3280MPS). This configuration will be maintained until old training loads are no longer required; at that time, an 1100/44, 8/32 IC's, and the base and IC select switch will be removed. The JSC Building 5 hardware will then present its final configuration as depicted in Figure 6.

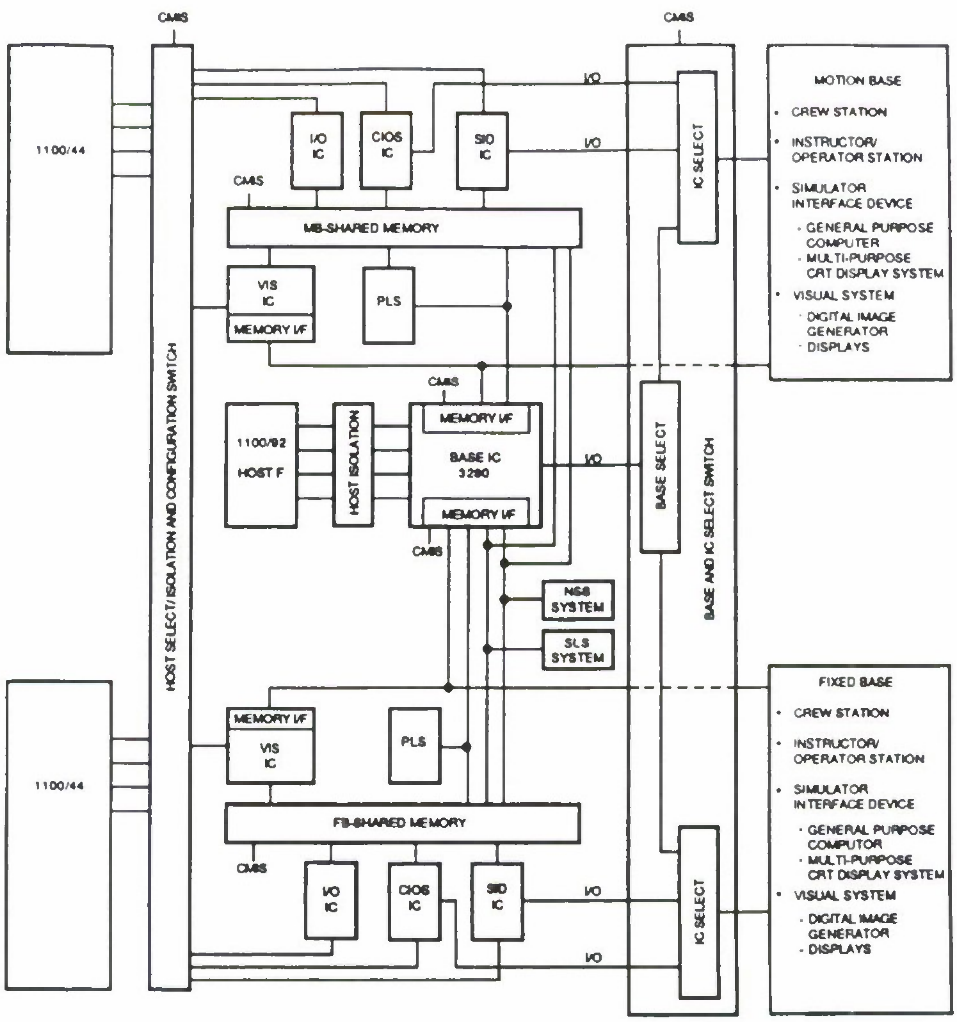


FIGURE 9 JSC BUILDING 5 UPGRADE CONFIGURATION PHASE 1

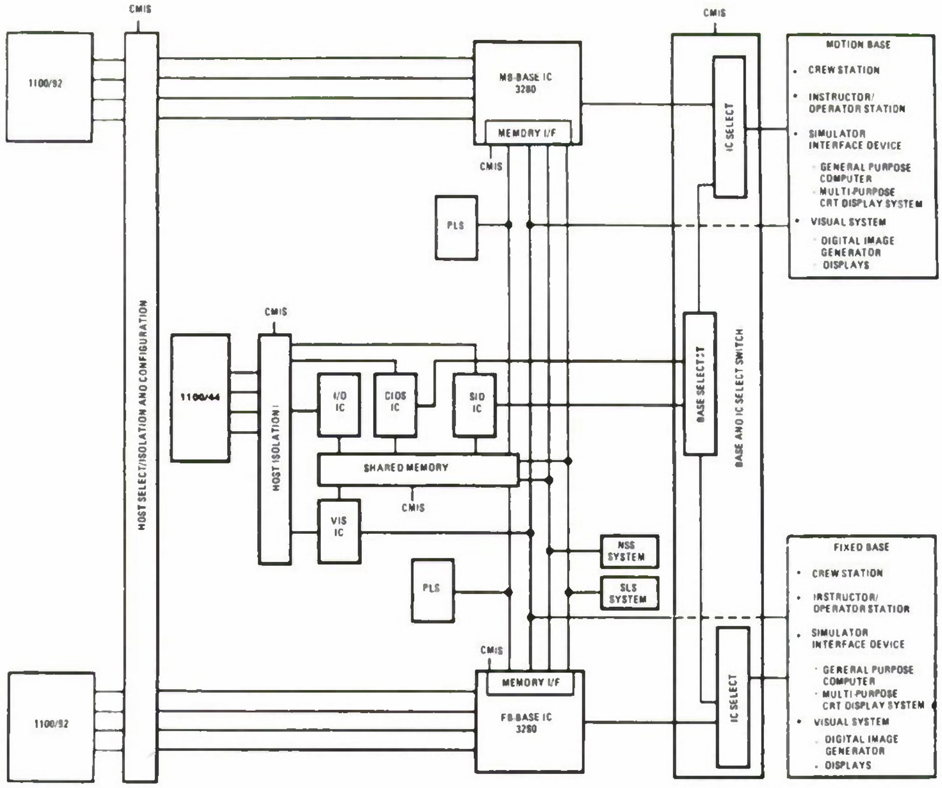


FIGURE 10 BUILDING 5 UPGRADE CONFIGURATION PHASE 2

Software Implementation Plan

The software implementation plan for this program follows the normal steps for simulator software implementation.

The initial step for software implementation is to establish and certify a software baseline. The operating system, real-time, off-line support, and applications programs are developed and integrated into the system in the order given. Each set of software must pass through a block test and master load sequence by passing predefined test requirements.

The interesting part of the plan is how to install the software changes, provided by the operations contractor, to the development baseline. The established baseline will have software updates delivered by the operations contractor on defined intervals during the development cycle. This software delivery is accompanied by documentation (design reviews, etc.) and test results. The documentation and tests are analyzed by the development contractor to determine the effect of software change on the system. If the analysis of the change affects the system, then a design and code walkthrough is held. The software is then taken through the normal block, test, master load sequence of all other software changes. Four months prior to the initial software delivery on the GNS, the baseline is frozen and recertified. Any nondevelopment upgrade changes after this time will be installed after final acceptance.

VERIFICATION AND VALIDATION

The SMTF Upgrade Verification and Validation (V&V) process is a structured testing approach to demonstrate that program requirements are satisfied and ensure that simulation functionality is maintained. The SMTF Upgrade program utilizes a bottom-up/build-up testing approach from the lowest to the highest level of the product tree.

The SMTF Upgrade development cycle consists of seven phases, from requirements definition to performance demonstration, as shown in Figure 11.

The requirements definition phase provides the data to produce the product definition tree. The program test tree is developed with the test plan during the preliminary design phase. These two elements set the stage for the SMTF Upgrade basic test approach (shown in Figure 12) and are a major program management tool necessary to control the testing requirements process.

Each product in the product specification tree (hardware, software, or facility) must be tested at least once during the testing process. The basic test tool allows for a logical test buildup and provides a check and balance system to ensure that each product is tested sufficiently and at each level (module, unit, subsystem, etc.). The testing process for this program has been broken into two levels defined as formal and informal testing. The formal test is defined as those tests run during the performance definition phase. This testing is the process used for customer acceptance and independent verification testing. The formal testing validates or ensures simulation functionality. Formal test categories include:

- Simulation performance
- Security
- System support
- Training load conversion

Informal (in-house) testing of hardware and software will occur at the unit, subsystem, system, and integrated systems level. A bottom-up approach of design and code walkthroughs, clean compilations, and off-line and real-time testing is utilized to ensure proper implementation of requirements. The results of informal testing are available for customer review, but are not the criteria for acceptance.

Hardware and software baselines will be established to validate the system during formal testing. Certification of the baselines produces benchmark performance data which establish the criteria for determining the SMTF functionality. This data will then be used in the analysis of the formal test results to ensure that the upgraded SMTF functionality requirement is met.

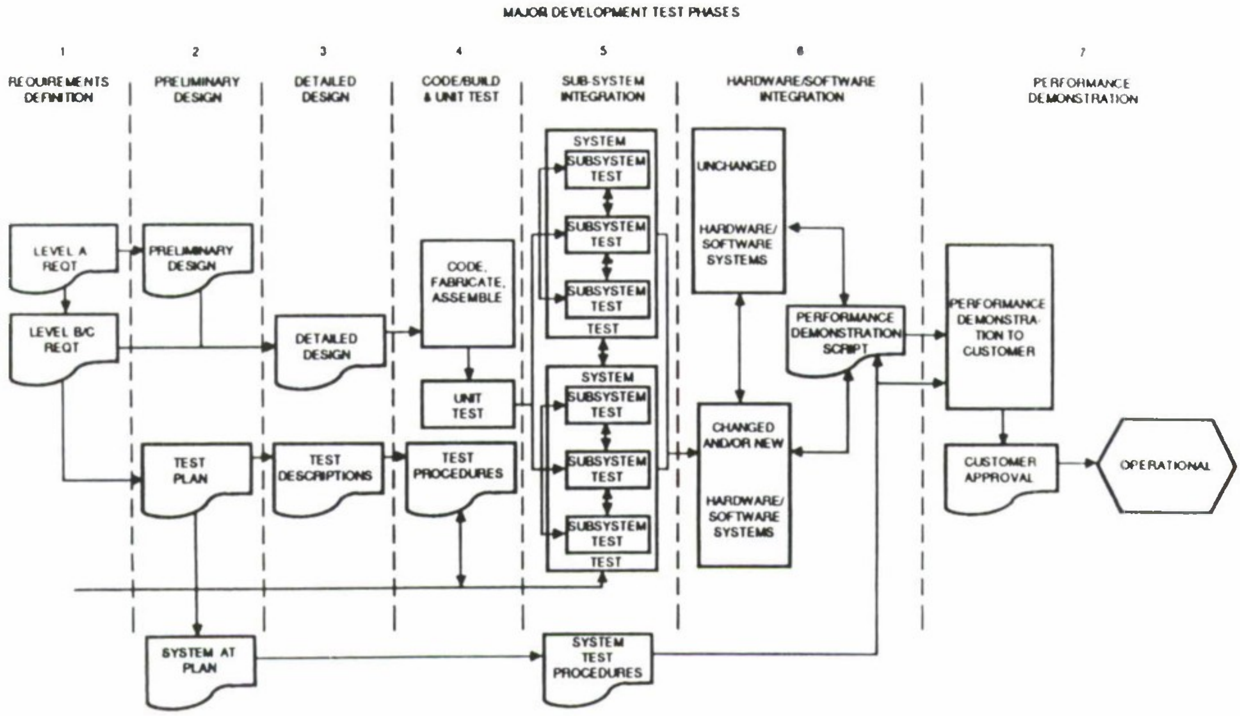


FIGURE 11 MAJOR DEVELOPMENT TEST PHASES

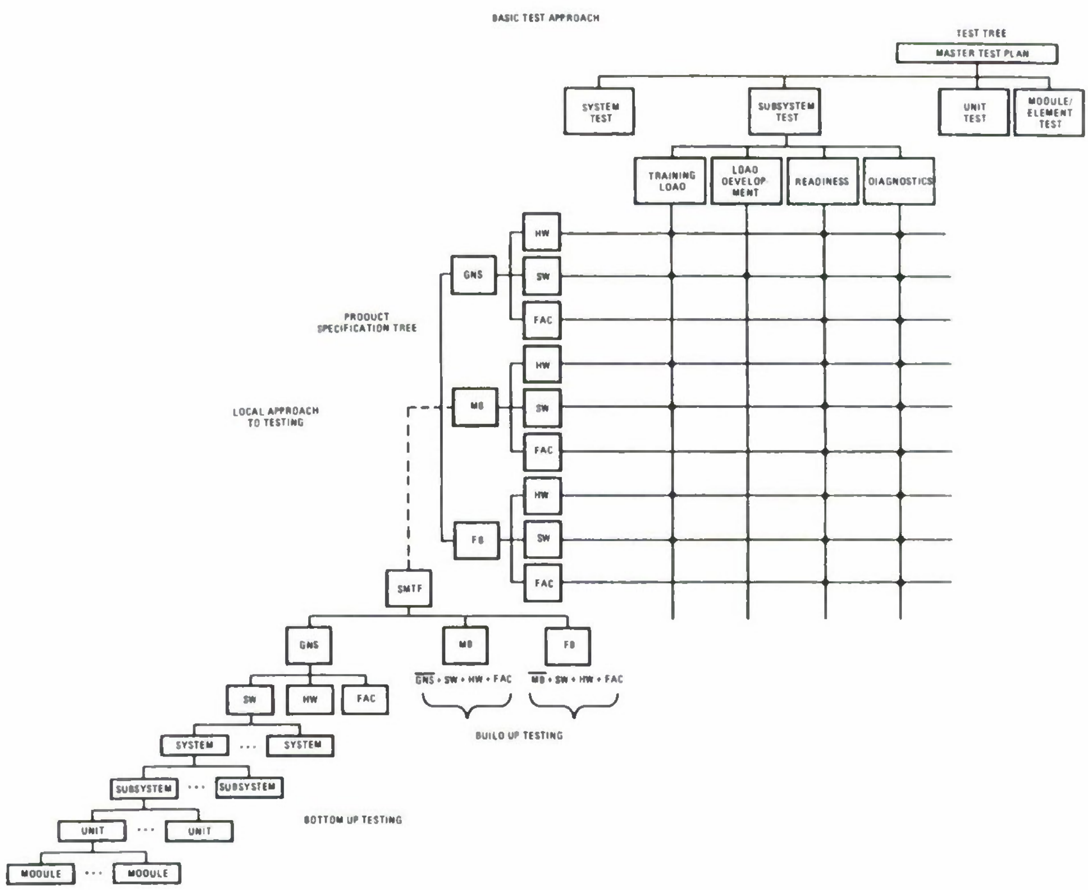


FIGURE 12 BASIC TEST APPROACH

CONCLUSION

The greatest challenges of this program have been in the hardware and software implementation phases. Having to maintain the training capability while installling each system added to the cost and complexity of the implementation cycle. CMIS control of the interim configurations lowered the risk of losing training hours during the development cycle. The logical approach to test provided complete and successful test results.

REFERENCES

JSC-22431	Johnson Space Center Security Requirements of the Shuttle Mission Training Facility Upgrade
JSC-22319	System Specification for the Shuttle Mission Training Facility Upgrade Step 1
JSC-22324	Facility Requirements for the Shuttle Mission Training Facility Upgrade Step 1
JSC-22350	System Concept Paper for the Shuttle Mission Training Facility Upgrade Step 1
JSC-20141	Security Requirements Plan for the Shuttle Mission Training Facility Upgrade Step 1
JSC-22397	Hardware Level B/C Requirements for the Shuttle Mission Training Facility Upgrade Step 1
JSC-22398	Software Level B/C Requirements for the Shuttle Mission Training Facility Upgrade Step 1
JSC-22485	Systems Test Plan for the Shuttle Mission Training Facility Upgrade Step 1
JSC-22483	General Design Document for the Shuttle Mission Training Facility Upgrade Step 1
JSC-22559	Shuttle Mission Training Facility Upgrade System Test Descriptions
JSC-22583	Shuttle Mission Training Facility Upgrade Software Detailed Design Document

ABOUT THE AUTHORS

Dr. Riley McCafferty is presently the General Manager for Space Programs Operations at the Singer-Link facility in Houston, TX. He has 35 years of technical experience, 26 of them in a managerial capacity within the aerospace industry. His educational background includes a PhD in Aerospace Systems Engineering from Clayton University, St. Louis, Missouri. Dr. McCafferty has had published many articles and papers ranging from "Man's Endurance in Space" to "Management (An Education Institution)" and has received numerous honors, awards, and commendations during his service within the aerospace industry.

Kurt Frevert is presently the Program Manager for the SMTF Upgrade. He has been involved in the SMTF since conception as a Design Engineer, Test Engineer, and Engineering Manager. His educational background includes an MSEE from the University of Missouri and a BS in Mathematics from Wartburg College.

ASAP & MANPRINT: WILL THE MARRIAGE LAST

Anne S. Sicilia
Advanced Technology, Inc.
Orlando, Florida

ABSTRACT

Over the years there has been a great deal of discussion about the length and quality of the acquisition process. Plans have been developed to improve the management of this complex process, sharpen its focus, and shorten the time it takes to complete it. Has this attention created a better mousetrap? Yes. Is there still room for improvement? Definitely yes.

This article addresses recent attempts to improve the Army acquisition process, the problems associated with the current process, and suggests that methodologies exist that can improve the acquisition process. These methodologies are the Army Streamlined Acquisition Process (ASAP) and the Manpower and Personnel Integration (MANPRINT) program. The premise of this paper is that when properly used together, the result will be an abbreviated yet more efficient time line.

INTRODUCTION

"The purpose of this article is to present some new and innovative ideas for shortening acquisition time." This quote is from an earlier article that addressed the length of the acquisition process. Unfortunately, despite the attention, and years of interest, there still exists a need for a shorter, streamlined acquisition process that works.

Maybe the lack of progress in shortening the acquisition process is related to the fact that the length of the acquisition process is only one part of a many faceted cycle. The set of problems listed below surfaced after World War II

Acquisition Problems

- Growing Defense Budgets
- More Complex Systems
- Decrease in Manpower
- Longer Acquisition Timelines

and still plague both DoD and industry acquisition procedures. Although it is generally accepted that the longer acquisition times imply a higher end item cost, the cost is not the only concern. Excess time in the acquisition process also increases the likelihood that there will be adjustments in technology and/or changes in the threat before the system is fielded. Without time constraints and adequate information inputs that address all of the acquisition problems, the length of the acquisition process will continue to increase with the results being more costly, less effective systems being fielded.

A valid approach to streamlining the acquisition process has to involve more than just eliminating time and activities. To be successful, it must be based on using information that is available before program initiation. Knowing, recording and updating data available from front-end analyses will support the streamlining process. The ASAP is a good start on shortening the process, but it lacks the integration techniques necessary to make it a success. MANPRINT with its focus on combining manpower, personnel, training (MPT), health-safety, and human factors engineering can supply the missing link to better define what is needed and when it is needed. The balance of the paper will discuss some attempts that have been made to shorten system acquisition and suggest that an integration methodology, like MANPRINT, is what is needed to make an abbreviated cycle work.

WHAT HAS BEEN DONE

1981 Time Period

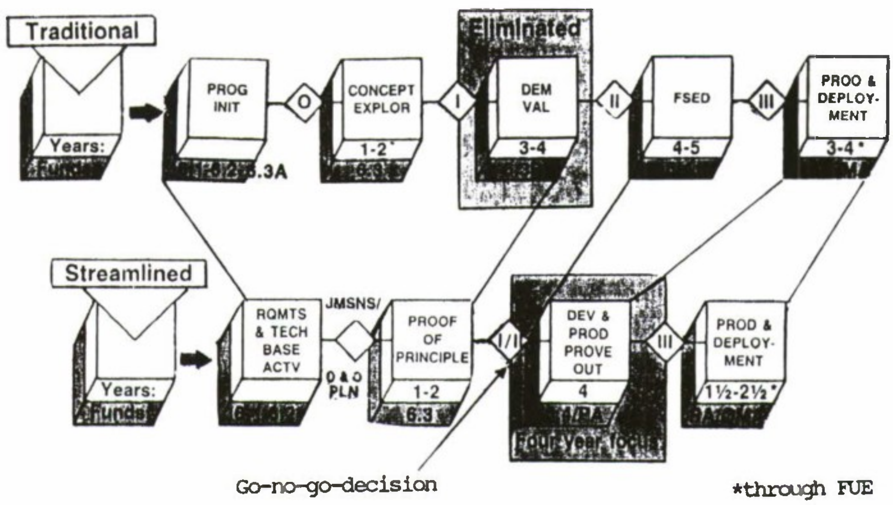
Frank Carlucci, as Deputy Secretary of Defense, set in motion a review of the acquisition process based on better management techniques. He tasked industry to contribute by designing the best, least complicated operating and support features. DoD worked with industry to review instructions and directives that had accumulated over the years, so that unnecessary steps could be eliminated. This review of the entire defense acquisition process culminated in thirty-two initiatives. Some of the initiatives are highlighted here.

Actions to Improve the Acquisition Process

1. Management Principles
2. Preplanned Prod: Improvement
3. Encourage Capital Investment
4. Front-End Funding
5. Standard Operational/ Support System
6. Funding Flexibility
7. Government Legislation
8. Increase Competition

These and the other initiatives supported the idea that the acquisition process needs to be a team effort between DoD and industry. Carlucci's plan stressed management and operational functions. His review was based on the realization

Acquisition Process Comparison



that the increase in the Soviet threat necessitated a rebuilding of basic defense industries. His initiatives focused on management, but did little to shorten the acquisition cycle. The plan likewise did not appear to adequately address the design considerations needed to compensate for operating and maintaining more complex systems with fewer, less skilled soldiers. Thus, Carlucci's modified acquisition process was a point for departure, rather than the final solution.

Army Streamlined Acquisition Process (ASAP)

Many of those goals have been carried forward to become the starting point for the ASAP. The ASAP as an acquisition methodology expanded on Carlucci's plan and defined demanding timelines. The objective of the process is to provide industry with greater flexibility in determining how best to meet Army materiel requirements. Skeptics say that it is just another attempt at modifying the process and will simply disappear, again with no visible signs of improvement remaining.

In defense of the ASAP, it uses the Carlucci plan as the point for departure and continues the delegation of management responsibility and accountability. It encourages managers to focus on what technology is available or scheduled to be available when a system is fielded. They are encouraged to spend time in the planning stages defining and clarifying the deficiency before a program is initiated. The ASAP invites industry to become involved much earlier in the acquisition process and to help solve the problem rather than just provide a product.

In relation to the Traditional Acquisition model, the ASAP is able to tailor the previously lock-step acquisition process. The Acquisition Process Comparison displays a simplistic comparison of these processes. The ASAP consolidates the Concept Exploration and Demonstration-Validation phases into the Requirements/Tech Base Activities and Proof of Principle phase. This approach permits greater RDTE funding flexibility by removing the artificial barriers (6.2A and 6.2B funds) between nonsystem and system-related advanced development.

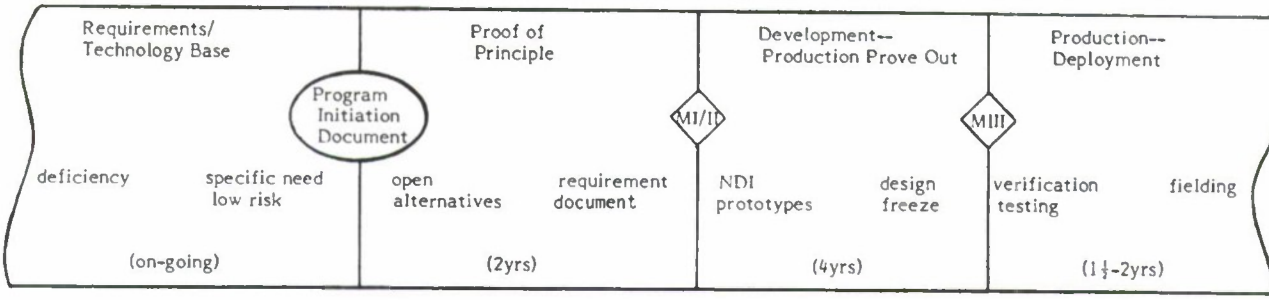
During Proof of Principle the ASAP gains insight into the maturity of the technology, as well as the operational concept, when a brass board prototype system is demonstrated by user troops. This user test also enables MANPRINT to assess the impact of the soldier requirements before a final design is selected. Alternative solutions are still possible at this point in the acquisition process. The system will not advance further if a high risk technology, manpower shortage, or other problem is identified during tests.

A successful demonstration of associated technologies and an approved requirement document, e.g., Required Operational Capability (ROC), ends the Proof of Principle phase. This constitutes a total commitment by the Army to fully fund and field this system and marks the beginning of the Development-Production Prove Out phase. Once the system concept progresses this far, all proposed changes to the design must be challenged. Test-analyze-and-fix (do it now) is key to this phase. Program stability is required from here to fielding in order to move rapidly towards and through the Production Deployment phase and deliver the system sooner to the soldier.

RESULTS THUS FAR

The ASAP is continuing to gain momentum and support. Activities within the acquisition process are trying to react to these quicker process time constraints with varying levels of success.

Army Streamlined Acquisition Process (ASAP)



Subsection (a) of Section 2434 of title 10, United States code (as designated by section 101(a) of the Goldwater-Nichols Department of Defense Reorganization Act of 1986) stated that before approval of a major system, manpower estimates must be presented to Congress prior to full-scale engineering development or production and deployment of that major system.

These actions support the hypothesis that the ASAP and MANPRINT are making a positive impact on the acquisition process.

Several concerns remain:

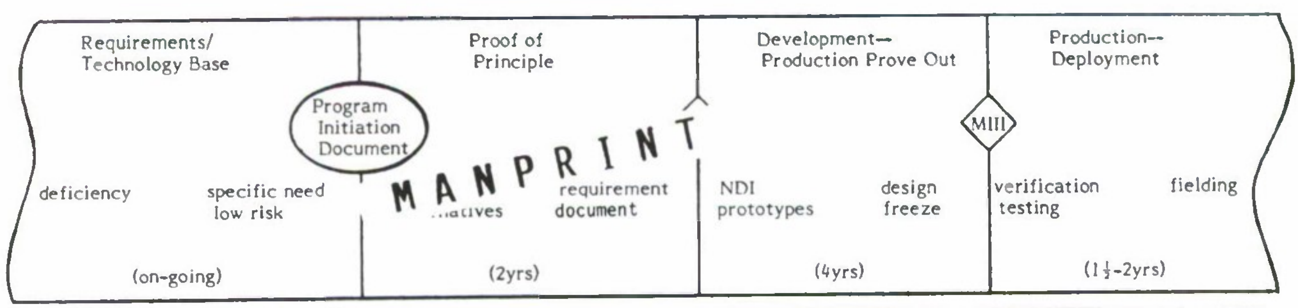
The process is still operations oriented
Manpower estimates are not due until after Army commitments to support the system concept is made, and

Other MANPRINT domains, e.g., safety, human factors engineering, are not included in the Congressional approval procedure.

The previously mentioned Defense Reorganization Act set manpower data requirements. The shortfall in the Act is the time period required for the reporting and approval of that data. Recall that in the ASAP, the commitment to fully fund a system is made at end of Proof of Principle. This equates to at least two years' development time and dollars before the FY87 Act calls for a manpower estimate. This can mean a great deal of resources expended, and halted, if a major manpower impact is reported. Chances are that after six or more years into the acquisition process, the system under development would be continued in hopes the people-system would be able to cope.

The FY87 Act, as with Carlucci's acquisition model, is a good idea and can be classified as either another innovative idea or as a point for departure. It is not the final answer.

Army Streamlined Acquisition Process (ASAP)



WHAT CAN BE DONE

MANPRINT. Let the integrating baseline of MANPRINT start early in the material acquisition process, prior to program initiation. Right now MANPRINT has the visibility and supportability that is required to make this happen.

MANPRINT guidance begins before program initiation when a deficiency is identified. In most cases there are either entire predecessor systems or components in which to draw information. It is the MANPRINT manager that sets up the management plan at this early point in the acquisition process. Although the answers may not all be available, the points of contacts and questions are. This period is still within the Requirements/Tech Base Activities phase of the ASAP.

During Proof of Principle the deficiency requirements are solidified. Again MANPRINT is able to keep track of MPT, safety-hazard and human factors environment implications that have been defined to this point. Testing and resulting data can be evaluated and used to assist with the materiel selection process. At this point MANPRINT people-constraints assist in defining or eliminating alternative selection possibilities.

When the requirement document, e.g., ROC, is developed, high driver (big problem) Military Occupational Specialties can be identified by either the predecessor descriptions or by reviewing available manpower authorization tables. Utilizing MANPRINT for its integrating values, a manpower estimate could be made to Congress when the Development-Production Prove Out phase is entered. Since alternative solutions are still being considered, this might be a system/manpower band rather than a set of single values.

By the end of this phase a much more definitive number could be presented for Congressional approval. The validity would be much greater than the widely used "MANPOWER estimate = NO IMPACT" statement. Although this statement is popular, it is also the cause of design failures when the system is fielded to a command with an inadequate MOS structure.

So what can be done? Streamline the acquisition process with a flexible and tailored methodology.
Overlay an interactive, integrating MPT oriented plan. Then create a system for the soldier that enables him to maintain readiness and successfully counter a wartime threat.

SUMMARY

ASAP has the time constraints identified and defined for a shortened acquisition process. It is time for this procedure to be MANPRINTed.

There has been too much talk by both the MPT and weapons systems communities about the length of the acquisition process. It is the quality of the resources spent during ASAP that need to be addressed. The youth population decrease and limitations on highly skilled individuals are now part of our data bases. If the people system is inadequately addressed or addressed too late, there is a chance that a multi-billion dollar, state-of-the-art weapons system will be fielded in eight years, and sit unused. Reason: no one is able to operate or maintain it.

MANPRINT has the potential to offer this people-system integration mechanism. It needs to be, as Carlucci foresaw the acquisition process, a team effort between DoD and industry. The acquisition players have all the pieces available. The factors only need to be identified, recorded, updated, and fed back into the system (weapons and people) development process.

REFERENCES

Making the Acquisition Process Efficient. Frank C. Carlucci, Deputy Secretary of Defense, Program Manager, Vol X, No. 5, Sep-Oct 1981

Another Look at Shortening Acquisition Time. John R. Snoderly, David D. Acker, Program Manager, Vol. X, No. 6, Nov-Dec 1981

Army Regulations AR 70-1, Sys Acq Policy and Procedures. AR 602-2, MANPRINT

A New Agenda for Manpower Analysis. LTC Wm Blackwood, MANPRINT Bulletin Vol. 1, No. 10, Apr 1987

Excerpts from Materiel for Winning Mar 1987 AMC/TRADOC

The Acquisition Process. LTG Robert L. Moore, AMC Deputy CG for RD&A, Army RD&A Magazine, Sep-Oct 1985

ABOUT THE AUTHOR

Anne Sicilia is currently a Management Analyst with Advanced Technology, Inc., and is supporting the Army effort to integrate ASAP and MANPRINT as they relate to ongoing and planned Army projects. Ms. Sicilia previously worked for the Project Manager for Training Devices (PM TRADE) and concentrated on Army policies, implementation strategies, regulations and procedures for the ASAP and MANPRINT processes. Before moving with her family from Washington, D. C. in 1984, she was employed by the Center for Naval Analyses (CNA). Efforts there focused on training and support available for the Naval Resource forces.

DETERMINING THE IMPLEMENTATION COSTS AND BENEFITS OF AN AUTOMATED TRAINING SYSTEM: PROBLEMS AND SOLUTIONS

John J. O'Connor
VERAC, Inc.
Air Force Human Resources Laboratory/T1
Bergstrom AFB TX 78743-5000

and

Capt Martin W. Pellum
Air Force Human Resources Laboratory
Brooks AFB TX 78235-5000

ABSTRACT

This paper addresses the cost-benefit dilemma encountered with the Air Forces' Advanced On-the-job Training System (AOTS) prototype development effort underway at Bergstrom AFB TX. Since AOTS is a <u>prototype</u> computer based training system, there is much to be decided before a final operational deployment configuration can be derived. The dilemma involves the trade-off between alternative computer architectures and the resulting training benefits each alternative will yield. The paper discusses the development of a microcomputer based cost model to assess alternative deployment scenarios of the AOTS into the operational Air Force. A methodology to assess the benefits derived from an AOTS implementation is presented. The description of the methodology includes a description of surveys and questionnaires that were administered to representative Air Force enlisted members at operational bases.

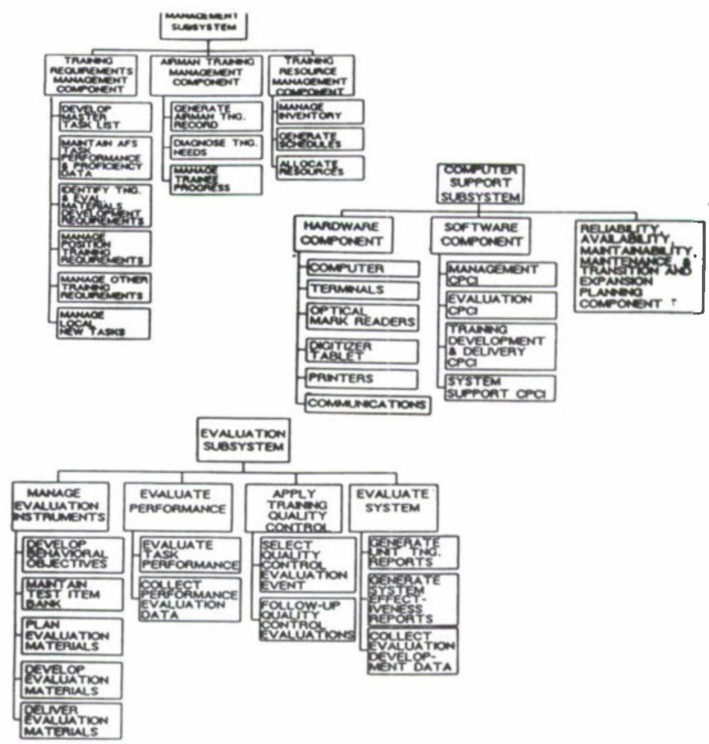
INTRODUCTION

Many advances have been made in the methods and technologies used for identifying training needs, developing training programs, managing training resources, and evaluating training effectiveness. Trends in computing power and data storage media have brought such advances as interactive videodisc and personalized training paradigms to a cost competitive level. The AOTS prototype is an attempt to combine some of these new training methods and emerging technologies. Primary goals of the AOTS include the attainment of a better OJT program by making more job-specific information available to training managers, trainers, and trainees; reduction of the administrative burden associated with Air Force OJT; and the implementation of computer based training techniques where feasible for job site training.

The AOTS prototype is a four year development effort underway at Bergstrom AFB TX. The major AOTS subsystems include: training management subsystem, training evaluation subsystem, and the training development/delivery subsystem. These functional subsystems are being coded in the Ada programming langauge to facilitate rehosting on a variety of host computer systems available to, or being acquired by the US Air Force. The computer support subsystem includes a Digital Equipment Corporation VAX 8650 with several upgrades at Brooks AFB, San Antonio TX. The Douglas Aircraft Corp. programming staff and the Air Force Human Resources Laboratory staff are tenants at Bergstrom AFB TX. The Bergstrom staffs are using Zenith Z-248 microcomputers linked via high speed fiber optic data lines to the VAX main frame at Brooks AFB.

This operational location is not customary for typical Laboratory funded R&D efforts. The design of this training system is being accomplished at an operational location with a large input from Air Force users to solve the real world Air Force OJT problems mentioned above. The Bergstrom AFB location also provides easy access to Air Force Reserve work centers to insure that the AOTS solves the OJT problems of the Reserves as well as the active duty forces. The specific needs of the Air National Guard are being addressed by the participation of the Texas Air Guard's 174th Fighter Interceptor Group at nearby Ellington ANGB.

Figure 1 shows the primary functions provided by the major subsystems of the AOTS prototype.



PURPOSE OF TASK

Plans to add an Advanced On-The-Job Training System (AOTS) to the USAF training process have led to numerous questions and issues related to cost versus effectiveness of such a system. The purpose of this task was to develop a model for estimating the costs of various AOTS concepts. The model is designed to estimate the costs of bringing the AOTS hardware and software (including courseware) on-line and maintaining the system for a specified number of years. The model provides the Air Force decision makers with a means to assess the costs of the "tools" for providing OJT to USAF enlisted personnel via an automated system. It does not provide a means of estimating the cost of actually training USAF personnel using AOTS.

Design Objectives

Although the model was not designed to explicitly address cost versus effectiveness issues, it has several important design objectives that allow an implicit comparison of cost-benefit alternatives. The model was designed to allow the user to rapidly produce first order cost estimates with minimum input data for a particular situation. This feature allows users, to identify "cost drivers" in varying deployment scenarios. When the cost drivers are identified, a focused study on benefits derived in these cost areas would be appropriate for a detailed cost-benefit analysis.

Another design objective of the model was to allow the affect of parameter uncertainties to be rapidly evaluated. Many data required to assess the costs of an operational AOTS are not known yet. In many cases, the model needs a "number" in the cost estimating relationships to run. Hence, the model was designed to run on a PC to avoid the longer turnaround times typical of main frames today. This design feature allows an easy way to converge on the unknown "number."

A key goal of the architecture was to provide an IBM PC compatible model using a commercial spreadsheet. The LOTUS version 2 spreadsheet software was selected because of its user friendly characteristics and widespread use.

Finally, the last design objective to be mentioned here was to provide a costing "tool" for Air Force use in preparing 5 year budgets for the POM process. The model described here is the first deliverable of an evolving tool for cost-benefit assessments, POM budget preparations, and other trade-off analyses. Many assumptions were made to keep the cost model simple, yet robust enough to represent the recurring and one-time costs of the job-site training environment.

AOTS COST MODEL STRUCTURE: OVERVIEW

"Costs" of complex systems are best accounted for in a structured method. A proven detailed Work Breakdown Structure (WBS) technique common in Air Force life cycle costing was chosen as the analytical method. The formulation of a detailed WBS has an impact upon model architecture and it specifies which cost-estimating-relationships (CERs) must be developed. Hence, the WBS and accompanying CERs provide a means of systematically estimating costs within the framework of the expected operational system. Double counting is eliminated and all essential cost elements are treated by using the WBS method.

The finished cost model is comprised of 5 separate Lotus 1.2.3 spreadsheets. First, the user is ushered through the input process in the first spreadsheet. This spreadsheet contains default values, but virtually any number can be changed by the user. These inputs allow users to describe training populations by MAJCOM by 2-digit Air Force Specialty (AFS), define the ratio of trainees per terminal (various types of terminals are offered), and define the character and quantity of "generic" Air Force bases. The input spreadsheet also allows the user to define several "constants" that are used throughout the CERs. These constants include costs per square foot for new facilities, average power consumption, costs per man-year for software maintenance, and many more.

When the user has completed modifying the input parameters, the output spreadsheet is called. This spreadsheet automatically calculates the CERs and summarizes costs by WBS item, by 2-digit AFSs, and by MAJCOMs. Another spreadsheet can be called to produce graphs of the calculated costs. Finally, a separate spreadsheet for POM budgeting and variations of some of the costing assumptions in the main model is provided. As with any simulation model, compromises and departures from "real life" are necessary. A special effort was made to make the AOTS cost model "User Friendly." Automatic calculations using LOTUS Version-2 MACRO commands allow users not familiar with the LOTUS software to use the model by means of the Menus provided.

MODEL INPUTS

The model was designed to run with a minimum of data inputs for each run. However, setting up the model for the initial run will require a substantial number of entries. To reduce this, default values were provided for all data categories based on Air Force costing elements, and, in the absence of better data, educated estimates of reasonable costs. Once again, <u>all</u> inputs can be changed by the user. There are very few "hard wired" parameters in the CERs.

General Air Force Data

The model developers characterized the training population by the number of trainees in OJT, and the distribution of the trainees throughout the Air Force.

Numbers of Trainees

Under the OJT concept envisioned under the AOTS, <u>all</u> Air Force enlisted personnel will be managed, trained, and/or evaluated by the use of various functional capabilities of the new system. Therefore, the total Air Force enlisted population was used as the "training" population for AOTS costing relationships. Data from the Air Force Military Personnel Center (AFMPC) were provided that categorized enlisted population by MAJCOM, and by 2-digit Air Force Specialty. After detailed discussions with USAF personnel, only 42 of the 55 two-digit AFS categories were included in the default data base. This reduction accounted for the AFS mergers underway or impending in the near future. Also, some AFSs are sufficiently small in population to skew the generalizations planned for this first-cut cost model. However, there is room in the model to add 13 additional AFS categories for future use.

Generic base definitions

A generic base concept is used to simulate the USAF base structure so that equipment allocations can be rounded-up to better account for the imperfect distribution of the training populations among the various bases in the Air Force. The generic base concept also allows for the accounting of existing base host computers (BHs) that may be available to specific bases and to differentiate between CONUS and Non-CONUS bases.

To better understand the round-up effect, consider a hypothetical situation such that all of the approximately 490,000 USAF trainees are located at a single base. In this event, hardware elements of the AOTS (ie., simple terminals to function as "work stations", elaborate systems to provide "training stations", interactive videodisc systems, etc.) could be distributed optimally among the AFSs, with at most one hardware element round-up per AFS. The perfectly distributed hardware elements at the single huge base would then fix the number of BH units that would be required with perhaps a single BH unit round-up required. If a single-base structure did exist, it would represent the minimum BH and hardware element requirements necessary to implement the AOTS for the approximately 490,000 Air Force enlisted personnel (the AOTS "trainees").

In actual fact, the USAF trainees are distributed among 125 primary bases, and the AFSs at each base are distributed non-uniformly. Hence, the round-up effect occurs at each base thus precluding optimal distribution of hardware elements and BHs among the trainees.

The generic base concept was adopted as a compromise between assuming perfect distribution of AOTS assets among the AFSs and attempting to model the "real world" USAF base structure. In the generic base concept, the 125 bases were grouped into seven generic types (Figure 2) according to

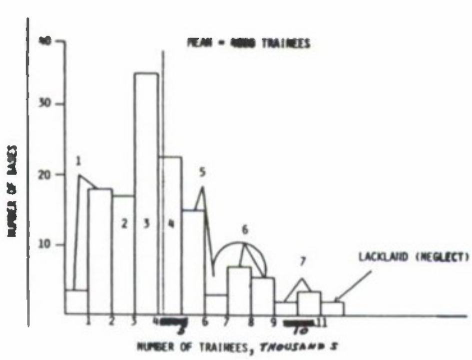


FIGURE 2. GENERIC BASES: 7 CATEGORIES BASED ON TRAINEE POPULATION

trainee population. The assumption was made that the AFS population was uniformly distributed across all bases. The analysis below describes how this approximation might affect the results at "real world bases".

Generic base analysis

As shown on the chart in figure 3, 18 bases and 24 AFSs (representing 90.1% of the trainee population) comprised the sample. These 24 AFSs were further reduced to four categories based on Air Force supplied subjectiva assessments of "complexity":

1. those that are highly complex and highly technical

2. low complexity and highly technical

3. highly complex but non-technical

4. and finally, low complexity and non-technical.

BASE	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
SAMPLE TRAINEES									
TYPE 1	2082	1521	1469	2616	1647	1901	1010	1330	[illegible]
TYPE 2	259	193	200	126	257	675	124	932	[illegible]
TYPE 3	259	206	362	234	[illegible]	267	226	176	[illegible]
TYPE 4	720	687	1100	594	1301	1241	550	1343	[illegible]
TOTAL	3320	2606	3291	2970	3407	4057	2729	3789	[illegible]
% BASE	91.6	82.7	88.4	87.0	90.4	95.4	90.4	90.2	[illegible]

BASE	DAVIS MONTHAN	EDWARDS	EGLIN	BROOKS	GRAND FORKS	MT. HOME	TINKER	WRIGHT PATTERSON	LOS ANGELES
SAMPLE TRAINEES									
TYPE 1	2904	2075	5064	2306	1020	2431	3473	1082	294
TYPE 2	297	273	452	95	245	144	1093	487	[illegible]
TYPE 3	323	211	449	57	207	200	250	301	[illegible]
TYPE 4	616	853	1569	318	1024	509	1009	1150	[illegible]
TOTAL	4140	3412	7534	2056	4176	3444	5741	3020	[illegible]
% BASE	82.9	89.4	89.7	86.1	87.1	94.8	92.2	78.8	[illegible]

FIGURE 3. 18 BASE SAMPLE

This sample set was then used to estimate possible errors in calculating the hardware requirements. Shown under each base name are the populations of the four AFS categories and the total number of trainees considered for that base. The % Base row is the percentage of trainees considered in this analysis relative to the total base population. The AFS populations at the sample bases varied from a low of 82.6% to a high of 95.4% compared to the 90.1% USAF trainee population encompassed by the 24 AFSs considered. Thus the sample demonstrated a reasonable representation of the population distribution across the 4 AFS complexity types.

Shown in Fig. 4 are the histograms for each of the four AFS catagories. The ordinate gives the number of occurrences of a particular percentage of base population from the sample shown previously. Note that some represent over-estimations, some represent under-estimations. From an overall standpoint, the USAF mean is centered about the histogram for all four trainee categories. This implies that about as many hardware requirements are underestimated as are overestimated; i.e., errors in one direction tend to be balanced by errors in the other. The actual error is a complex function of the maximum number of trainees on a particular base working a particular shift, number of trainees serviced by each hardware element and number of hardware elements that can be serviced by a particular BH configuration. From this short statistical analysis, the selected sample is shown

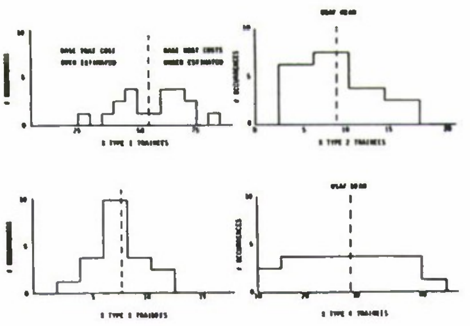


FIGURE 4. HISTOGRAMS OF AFS TYPES

to indeed be representative of the 125 base population for AFS trainees. Although the sample standard deviations are large, the errors incurred in the sample by assuming that all bases have the same distribution of AFSs tend to be balanced in terms of over- and underestimates.

Generic base categories

One goal was to assemble the best data base possible for use as default model inputs. To that end, the model developers accomplished an analysis of the 125 USAF bases relative to suggesting generic base categories. The results of this analysis were shown above in Figure 2. Plotted is the total number of trainees at a base (in a band of 1000) versus number of bases whose trainee population fall in that band.

AOTS specific data

AOTS specific data includes the number of work shifts typical of a specific AFS; hardware data, software requirements, and miscellaneous information regarding facilities, communications, etc. The model is AFS oriented in that production and maintenance costs for some of the hardware elements can be allocated to a specific AFS. The maximum number of trainees per shift is a necessary input to avoid buying AOTS hardware elements for off-duty trainees. The ratio of trainees per AOTS hardware element for each AFS is an important input since it sizes the Base Host and AOTS Central hardware requirements. Existing hardware may satisfy some of these requirements for an AFS and hence the model allows existing hardware to be input. Other inputs allow AFSs to have extra hardware elements to support their special needs.

AOTS specific hardware cost data is required for the AOTS Host Central (HC), Software Development and Maintenance Center (SWDC), fixed workstation (FWS), remote workstation (RWS), training station (TS), and Other Types (OT) of stations. Other data items needed include existing hardware and software requirements, including courseware.

LCC APPROACH BASED ON WORK BREAKDOWN STRUCTURE

The cost model is based on a Work Breakdown Structure (WBS) as defined MIL-STD-881, Military Standard Work Breakdown Structure for Defense Material Items. This standard defines general WBS structures for several different types of military systems. The standard reminds readers that all systems are unique and require adjustments to the general structure. The AOTS cost model follows the MIL-STD-881 guidance for Research Development Test & Evaluation (RDT&E) and Production phase costs. Since the standard does not address Operations & Support (O&S), DCA 600-60-1, Cost and Planning Factors Manual, the Joint Tactical Communications Office Life Cycle Costing Volume of the Cost Effectiveness Program Plan and other sources were referenced for WBS definition.

The LCC model has an extremely flexible data input structure. The hardware element (FWS, RWS, TS) requirements for each AFS can be treated individually or AFSs can be grouped into classes. The cost input parameters associated with the CERs can be easily changed in the Input Section of the model for "what if" analyses.

The model estimates total USAF Life Cycle Costs (LCC) in 1987 dollars according to the detailed WBS. The model assumes that RDT&E of the AOTS is completed in one year. Immediately following RDT&E, the hardware necessary to place the AOTS into service is procured during the Production phase, which also is assumed to require one year. The O&S phase is then assumed to last for ten years following the Production phase for a total life cycle of 12 years.

The POM model briefly mentioned above uses inputs from the basic LCC model to provide information regarding the impact of changing the acquisition times associated with RDT&E, Production, and O&S. This provides important planning information for the PPBS process. Also, if desired, cost impacts and economic studies related to discount and net present value assumptions may be accomplished using the POM model.

Cost Estimating Relationships (CERs)

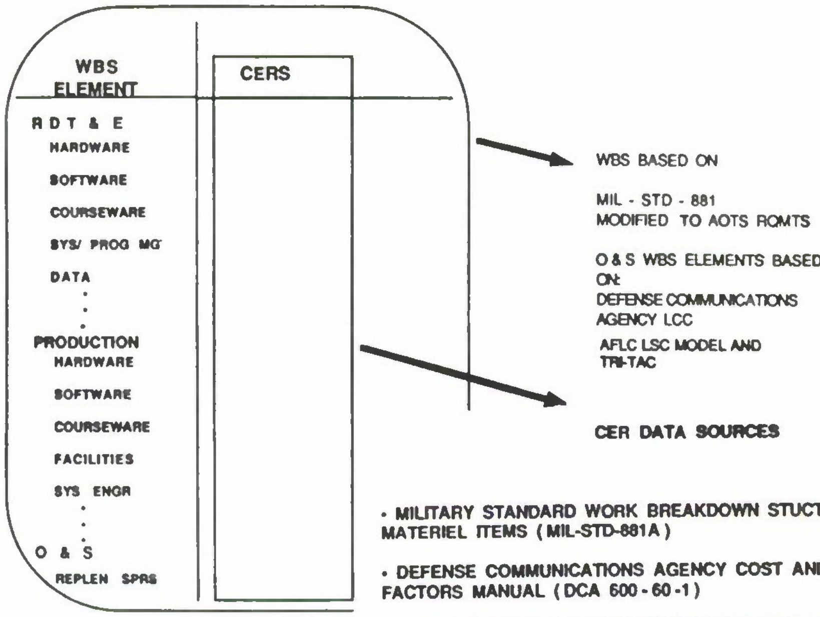
The CERs are mathematical expressions which relate AOTS parameters to program costs. Figure 5 shows example CERs relating the AOTS WBS parameters to the life cycle costs.

Cost model assumptions

The cost model assumptions establish a baseline and provide bounds for the modeling effort. All baseline costs are adjusted to FY87 dollars using Air Force Regulation 173-13. The model uses a 12 year life for the AOTS with one year for RDT&E, the next year for Production, and the next 10 years for O&S.

The model calculates maintenance costs assuming that maintenance is contracted to a commercial firm. The user can select between a budgetary cost estimate or an economic cost estimate. The difference between the two calculations is based on different AF personnel costs as defined in DCAI 600-60-1, DCA Cost and Planning Factors. The communication costs are assumed to be AUTODIN subscriber costs; special communication hardware costs are part of the individual hardware item costs and the

LCC APPROACH BASED ON WORK BREAKDOWN STRUCTURE



CER DATA SOURCES

- MILITARY STANDARD WORK BREAKDOWN STUCTURES FOR DEFENSE MATERIEL ITEMS (MIL-STD-881A)
- DEFENSE COMMUNICATIONS AGENCY COST AND PLANNING FACTORS MANUAL (DCA 600-60-1)
- OPERATING AND SUPPORT COSTS FOR COMMUNICATIONS SYSTEMS ANALYSIS AND RECOMMENDATIONS (INSTITUTE FOR DEFENSE ANALYSIS)
- OPERATING AND SUPPORT GUIDE FOR ARMY MATERIEL SYSTEMS (DA PAM 11-4)
- US AIR FORCE COST AND PLANNING FACTORS (AFR 173-13)
- SOFTWARE ENGINEERING ECONOMICS. (B. BOEHM)
- MODIA: VOL 5 A USERS GUIDE TO THE COST MODEL (RAND CORP)
- SATELLITE OPERATIONS COMPLEX INDEPENDENT COST STUDY DOCUMENTATION. (USAF/SD HEADQUARTERS)
- COST AND TRAINING EFFECTIVENESS GUIDE. (LITTON MELLONICS DIV)

FIGURE 5. CERS RELATING THE AOTS ARCHITECTURE PARAMETERS TO THE PROGRAM COSTS

COURSEWARE COST ESTIMATING RELATIONSHIP

RELATIONSHIP :

COST = (# CAI HOURS) X (SW DEVL HRS / CAI HR) X ($ / HR)

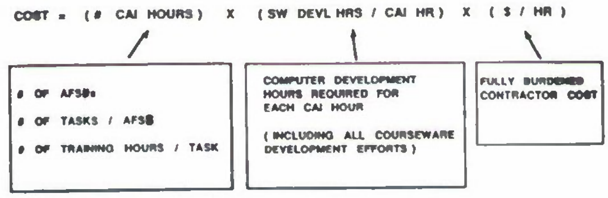


FIGURE 6 THE COURSEWARE CER

communication lines are shared by the host base and the Defense Communications Agency (DCA). The facilities and utilities costs are calculated for only the equipment purchased expressly for the AOTS. It is assumed that the hardware purchased for the AOTS program is purchased at GSA rates and that the vendor will deliver the items to the users' facilities.

Example: Flow of hardware element costs in LCC

The model has user specified values to calculate the number of hardware elements a base host computer can support. The default cost values delivered with the model use a ratio of 85 peripheral ports per base host computer. If the user adds additional trainees, or elects to separately purchase additional hardware elements so that more than 85 ports are required at a given base, the model will calculate the costs of an additional base host computer. The purchase of both additional hardware elements and base hosts increases the requirements for facilities and utilities, which the model automatically calculates.

An important model assumption is that existing hardware elements (base hosts, stations, etc.) do not add to the costs. That is, facilities, maintenance, communications of existing equipment utilized by AOTS is not an AOTS cost. Such costs will be funded by the original hardware user.

COURSEWARE COSTING APPROACH

Figure 6 shows a detailed example of the approach used to estimate courseware costs. The emphasis for the courseware CER development approach was to identify a relationship that would allow HRL to develop more accurate costs estimates as more data became available. A relationship was chosen that used courseware terminology consistent with HRL experience. No production phase cost elements are affected by a change in the courseware cost, since the definition of the courseware costs includes all efforts through the acceptance tests, which includes the production phase. In other words, the model does not presently allow the user to evaluate actual training costs by employing the courseware; only the cost of developing and maintaining the courseware (ie., the "tools" of computer based training) is presented.

DETERMINING THE BENEFITS OF A CHANGE
General Discussion

Managers invariably want to know the expected gain from any change proposed for the organization. Change may be in many forms. For example, integrating new technologies, redirecting work flow, altering policies or adding or deleting formal training programs. While a considerable amount of time is typically devoted to calculating the direct costs (manpower and equipment) associated with implementing such changes, much less time is spent assessing the impact (benefits and cost avoidances) these changes might have on the organization. Even less effort is directed toward measuring the impacts the changes actually have once adopted. The primary reason for this is no secret to program managers, particularly training directors--cost-benefit analysis is extremely difficult. Why? Because it requires a complete understanding of current organizational processes, how these could be affected by the planned change, how this effect can be observed, and how these effects can be translated into meaningful outcomes. In undertaking a cost-benefit study, there are at least three major questions that an analyst must ask.

Is Benefit Information Necessary?

It may be that costs alone are adequate for making decisions. For example, the company may not have the resources needed to buy the new equipment, no matter how nice or beneficial it would be. Thus, a complete assessment of the benefits would be of little value at this point in time. For the most part, however, an organization will have numerous alternatives, both in the level of change adopted and in the reallocation of resources to support the change. Such is the case for the AOTS, and so the need for a cost model to evaluate various implementation scenarios. But without benefit data, one might reject a program which, though more expensive than the alternatives, would have provided the greatest marginal gain.

What Information Should Be Gathered?

If cost-benefit data is needed, then several different factors should probably be measured. The implemented change itself is a logical starting point. Here the focus is on what the "change" entails; what would be done differently under the changed condition compared to what is done now. Having the baseline, or current practice, data is integral to calculating the relative impact of the planned change. However, this information may not be readily available for any number of reasons. It may not be routinely gathered, there may be a difference between what should be happening and what is actually happening out in the work force, or the planned change may be something totally new and without an apparent comparison practice. Nevertheless, the baseline data would need to be obtained prior to implementing the change.

A second factor to consider when determining what to measure is the ultimate goal of the change. Whereas the first factor discussed above focuses on what is to be changed, here the focus is on the desired effect of the change. In other words, the outcome. A single change will likely have several outcomes, depending on its nature. For example, it may impact individuals, groups, the organization as a whole, or all three levels. At a more micro level, particular individuals, groups or organizations may be affected differently than others. The outcomes that ultimately get measured are usually in the form of performance or output. As with the baseline information on what people do, there is often a lack of reliable information on how well people perform on the job. Although there does tend to be more data on group and organizational performance, these results are impacted by numerous other factors and constraints such that a single program change may not have any detectable effect.

A final major factor to consider in deciding what to measure is the reactions of the people affected by the change. There are basically three interlocking sides to this issue. The first is whether the people perceive the change as necessary. Resistance to change has caused many good programs to fail or not attain the expected level of effect. Some persons may see the program as holding potential benefit, while other may be more inclined to say "If it ain't broke,.don't fit it."

The second side is the willingness of the people to implement the change in the intended manner. Most managers have experienced the frustration of seeing new pieces of equipment (especially computers) intended to make the workers' jobs easier end up collecting dust because they are not used as planned. Measures of how the change actually manifests itself in the operational environment are therefore necessary.

The third side to this issue is the perceived effectiveness of the change. There may be times when the mere perception that a program is effective is enough to bring about an increase in production or performance. This placebo or "Hawthorne effect" can sometimes account for a considerable amount of the variance between the pre and post-change conditions. On the other hand, the people may implement the change properly in the short term, but see the change as having no immediate meaningful effect. In either case, the effects of the planned change will be short-lived and the results of the cost-benefit study could lead to potentially erroneous conclusions about the utility of the change.

How Should The Data Be Obtained?

Once benefit data is deemed necessary and defined, the analyst must decide how the critical factors are to be measured. Obviously, one thing that could be done is to estimate the outcomes in the absence of "real" data. This may be the best that can be done when one is dealing with something quite different from the current methods, like the AOTS is in the training sense. Cost estimates can sometimes be made rather accurately in this way, since they tend to be rather stable (e.g., the price of a computer will be about the same across situations). However, benefits and outcomes lack this stable cross-situational consistency. As a result, such estimates can emerge as unreliable when made in advance and without any data to back them up.

A popular method for obtaining the needed information is to conduct surveys and interviews of the persons affected by the change. Such methods are often used to help establish the baseline information (what is done currently). But they can also be used to collect benefit estimates from job incumbents even before the change is implemented on a test basis. They were used in both ways in the AOTS benefit study. Surveys are not without their problems. It is sometimes difficult to explain the change clearly enough to enable incumbents to estimate the effect it will have on their jobs or behavior. They may be suspicious about the purpose of the survey and may be reluctant to provide honest answers, or they may answer in a way that they feel is desired, or not desired, by the analyst.

A third way to obtain information is by going to the work site and observing what people do on the job. This can often be the most insightful and informative way to determine where the benefits of a planned change should be targeted.

A final method that can be used to gather benefit-type information is to use data already gathered for other purposes. Changes in such trend data as supply levels, repair times, or quality control inspection results could be attributed to the implemented change. Though it is these group or organizational outcomes that are most often the target of the change, they are affected by numerous other factors like availability of parts, the weather, or changes in evaluation standards. Thus, the analyst must consider these alternative explanations when using existing or routinely gathered data in making statements about the effect of an intervention.

DETERMINING THE BENEFITS OF AOTS

In considering the above questions with regards to the AOTS project, it became clear that indeed, benefit information would be needed if a full understanding and assessment of the AOTS is to occur. What to measure had to be decided upon. As with most cost-benefit analyses, it was desired that the benefit information be along the same lines as the cost information so the two could be compared more easily in determining the marginal returns from an AOTS investment. This meant that the benefit data should be convertible to dollars where possible. A review of training cost approaches used in the Air Force today revealed that most centered on the amount of time devoted to training activities, since time can be expressed in terms of pay for given individuals. Thus, time spent on OJT became one of the primary factors to be included in the AOTS benefit study.

As discussed in the preceding section, numerous other areas need to be considered when examining the effects of new technologies or programs. The perceived need for the AOTS, what certain aspects of the AOTS would be of value, and whether incumbents feel the AOTS would lead to improvements in the OJT process were all seen as issues to be included in the benefit assessment.

It became apparent early in the study development that the baseline information on OJT that would serve as the point of comparison for the estimated impacts of the AOTS, was not available. This meant that the decisions of what to measure had to be couched in the ability of the team to actually gather that information. Surveys and interviews were selected as the methods to be employed since they appeared the easiest to construct, could be administered to large groups of airmen, and could be analyzed fairly easily. It was at this point that the Air Force Research Laboratory's Personnel Survey Research Branch was asked to assist in developing the instruments and gathering the data.

Several rather distinct groups of people are involved in the OJT process. Since these groups differ in their roles now, conceivably the AOTS would impact them differentially. Therefore, four separate surveys were constructed, each targeted to a specific group of people: (1) squadron commanders, (2) base and unit OJT managers, (3) supervisors and trainers, and (4) trainees.

There were three basic thrusts reflected in the surveys and interviews. First, to gather baseline information about how OJT is done today in the Air Force. This would include the amount of time it takes on the part of OJT managers and supervisors/ trainers and how this time is distributed across specific training-related functions. Second, to obtain estimates of how an automated training system like the AOTS would impact the amount of time spent on training overall and/or on the relative time devoted to particular training areas. And third,

to determine the general contention toward AOTS by the different groups of people involved in the OJT process. In particular, what specific aspects of the AOTS would be of value to them and whether they believe AOTS would have a positive effect on OJT.

The Squadron Commander Survey

Squadron commander's are ultimately responsible for the OJT within their unit. They insure that a sound OJT program exists, it is carried out according to governing regulations, it results in quality performance, and that action is taken when individuals fail to meet established time lines for completing their training. For this reason, inputs from commanders were seen as potentially lending valuable information on how OJT is currently conducted throughout the Air Force and insight into how the AOTS could facilitate the process.

The commander's written survey addressed their perception of the importance of training to such things as "initial decisions to enter the Air Force", "safety", "job satisfaction", and "mission accomplishment". Since the role the AOTS might play in the commander's management of his/her squadron's OJT program was somewhat unclear, individual interviews were held that went considerably further in depth on how OJT is currently carried out in the squadron, what areas were in need of improvement, and how an automated system like the AOTS could help. The interviews focused on how the commanders felt they could benefit from the AOTS. For example, whenever an individual fails a quality assurance (QA) inspection, the commander must initiate an investigation to determine why. A review of the OJT records is one of the first orders of business to see if the cause could be the result of a training deficiency. During the interviews, commanders indicated that the AOTS could speed up this process considerably, especially if it also contained information about previous QA results.

The OJT Manager Survey

The base OJT manager is assigned to the Consolidated Base Personnel Office (CBPO). He/She coordinates all the unit OJT programs and assists the unit-level OJT managers. The unit OJT manager usually works directly for the squadron commander and so acts as the commander's eyes and ears regarding the status of the unit's OJT program. This person may be a full-time OJT manager or may have the responsibility in addition to their regular duties. The job of the OJT managers at both the base and unit levels center on the administrative aspects of the OJT, e.g. documentation, training time lines, and written testing. This is a rather simplistic view of the OJT managers' job, but it serves to indicate the area the surveys and interviews sought to get information about.

Training experts within the Air Force Military Personnel Center (AFMPC) assisted in classifying a typical OJT manager's job in terms of eight distinct functions. These functions were (1) managing the upgrade training program, (2) conducting/ reporting staff assistance visits, (3) managing the Career Development Course program, (4) coordinating with CBPO and outside agency functions, (5) assisting supervisors/commanders, (6) organizing/conducting meetings, (7) daily supervisory activities, and (8) administrative activities.

During the instrument development phase, the list of functions was presented to about ten training managers who found them to be rather complete in capturing the essence of the OJT managers job. Definitions for each function were given in the survey and during the interviews to insure that each manager understood the specific tasks encompassed within the particular areas. This was very important since the majority of the questions were referenced against these eight functions.

OJT managers were first asked to indicate which of the eight functions they felt was the most important, the least important, the one they would spend more time on if they could, and the one that needed the most attention or additional resources. The next section of the survey concerned the time currently spent in each of the eight areas. The managers first indicated what percentage of their duty time was devoted to training-related activities overall. They were then asked to describe how this training time was allocated across the functional areas, the sum being 100%. Finally, after reviewing a written description of an automated training system (it was not termed the AOTS within the survey to avoid any preconceived notions the managers might have had), they indicated how much each individual percentage would change, i.e., go up or down and how much, to correspond to how they felt their time spent on certain tasks might change as a result of having an automated system available.

A final section of the survey asked the OJT managers to indicate which aspects of the automated training system, as described in a written scenario, would hold the most value to them personally in their current job. The eight areas highlighted were: (1) training histories on individual airmen, (2) more task-level information than provided in current training standards, (3) standardized training practices, (4) standardized evaluation procedures, (5) automatic documentation of training, (6) providing training on some tasks via the computer, (7) providing evaluation on some tasks via the computer, and (8) automatic scheduling of training.

The Supervisor/Trainer Survey

Supervisors are the ones most directly responsible for insuring their subordinates are trained. In most cases, the supervisor, i.e. the person who formally evaluates another's performance, will also be the trainer. However, other persons are often designated by the supervisor to be trainers. This may occur for several reasons; perhaps to give the trainer some supervisory-like experience, the trainer may be a co-worker and so more directly attuned to the specific task requirements, or the supervisor may simply be too busy to devote the time necessary to one person's OJT. Nevertheless, the target group for this study included those who actually conducted the OJT, therefore the surveys were labeled for supervisors/trainers.

The structure of the supervisor/trainer surveys and interviews paralleled that of the OJT manager surveys. Again, the training experts at AFMPC were helpful, this time in

classifying the trainer's job in terms of its major functional areas. These, in turn, were presented to about 20 non-commissioned officers (NCOs) with experience at conducting OJT, who verified them as a reasonable breakdown of a trainer's responsibilities. The NCOs also reviewed the written description of the automated training system to insure it would be understood by similar NCOs in the field.

The seven functional areas covered in the supervisor/trainer survey were: (1) providing training or selecting trainers, (2) evaluating performance, (3) managing career development courses, (4) developing training program, (5) coordinating with OJT manager, (6) counseling trainees, and (7) documenting training.

Respondents indicated what percent of their duty time was devoted to training-related activities, how this time was distributed across the seven functional areas, how this time might adjust given an automated training system capability, and what particular aspects of the automated system were seen as being most valuable to them (the same ones as presented to the OJT managers). Additional information was gathered on how they identified an airman's training needs, how performance is evaluated prior to certification on a task, plus general background data, e.g. how many persons they provide training to, about how long it takes their trainees to become duty position qualified, and whether or not they routinely use computers in their own jobs.

The Trainee Survey

The last target group was trainees. This included persons currently in training or who had completed training within the last three years. There are several sides to OJT, and who is considered a trainee changes depending on which focus one is taking. In the most general sense, OJT is the training that one receives that leads to proficiency on those tasks that make up a particular job. By this definition, almost everyone, no matter how long they have been in the Air Force or what their rank is, enters OJT whenever they change jobs or when the tasks that entail their current job change. The AOTS is designed with this job qualification notion in mind.

There is another concept of OJT in the Air Force that centers on the training leading to an increase in the skill level designation for a person. Skill level is a critical factor in determining whether a person is eligible for promotion, a job change, and even if they can remain in the service. This upgrade training is made up of two components: position qualification training (as defined above) and career development course (CDC) training. A CDC is a correspondence course that covers a career specialty. Thus, their purpose is to give the trainee an understanding of the entire career field, not a specific job. To be upgraded to the next skill level, a trainee must complete both the requisite CDCs and become certified on all the tasks required of them in their current job.

AOTS is intended to incorporate both aspects of present day Air Force OJT. Unfortunately, position qualification under the current system is not tracked very well nor reliably. So in order to more clearly define the trainee target group in this study, upgrade training was selected as the type of training that the person should be currently engaged in or recently completed, rather than position qualification alone. For ease in identifying persons in this catagory, the definition of upgrade training was further restricted to include just those persons going from the 3-level (apprentice) to the 5-level (semi-skilled). For all practical purposes, this meant persons who had bean in the Air Force between 6 and 36 months.

The trainee survey included questions about how their trainer evaluated their performance prior to certifying them on tasks, what their primary source of job information was, how frequently their supervisor reviewed their training record with them, and how they would rate their OJT experience overall. Also gathered was background on their current training status (whether in training on job tasks, CDCs, both, or neither), how long it took them to become certified as proficient on their job tasks and/or complete their CDCs, and if they have had received any training via a computer.

DATA COLLECTION AND ANALYSIS

Twenty-four Air Force bases representing the five major operational commands (Air Training Command, Military Airlift Command, Strategic Air Command, and Tactical Air Command) were selected to participate in the written survey. Most of the OJT that is conducted in the Air Force occurs in these commands. All the bases were located within the continental United States.

The surveys were directed to base-level personnel since this is where the vast majority of the OJT occurs. The supervisor/ trainer and trainee target populations were restricted to the four Air Force specialties within which the prototype AOTS is being developed. These were Personnel, Security Police/Law Enforcement, Jet Engine Mechanic, and Aircraft Maintenance. This was done for three reasons. First, these specialties are diversified in the type of OJT that is typical across all Air Force specialties. They also are among the largest career areas in the Air Force and so carry much of the OJT load. Second, since the AOTS development has centered on them, more information is known about these specialties than others. NCOs from each are also on the AOTS development staff and so were readily available to aid in constructing and interpreting the surveys. And third, the information uncovered with the surveys could potentially help direct data gathering during the one-year operational test and evaluation of the AOTS at Bergstrom AFB. The test would, of course, be within the four specialty areas.

Approximately commander surveys, OJT Manager surveys, supervisor/trainer surveys, and trainee surveys were distributed. Two bases failed to receive the surveys while one failed to return them. The adjusted return rate was about %.

The surveys were mailed directly to the Survey Control Officer within each base's CBPO. This person typically worked with the Base OJT Manager to identify participants and distribute the surveys. Selection of the participants was made at

the base level and not by the study group. This was primarily because reliable information on who was in training and who was their trainer was not available in central personnel files. Wherever possible, actual pairs of trainers and trainees were to be selected so their responses could be compared. Through coding of individual answer sheets, the trainer and trainee responses to be matched during analysis without knowing the identity of either party. Marked answer sheets were entered into the data base by computerized optical scanners. Comment sheets were attached to each survey booklet to allow written statements.

One-on-one interviews were conducted by one AFHRL representative, one person from the AFMPC Survey Branch at six Air Force bases, and two from each of the three major operational commands. The interviews were done to help validate the written surveys and to uncover additional information not likely to emerge with the surveys or too difficult to put in the survey question and answer format. Each interview involved a person from one of the four target groups and followed the basic structure of the written surveys. Responses were coded onto the same type of answer sheet as used for the written surveys so the data could be merged prior to analysis. Antidotal information was also recorded, but not on answer sheets. A total of 115 interviews were conducted over a two week period.

The heart of the cost analysis will be in calculating the dollar value of any reduction in the amount of time OJT managers and supervisors/trainers spend on training-related activities. To accomplish this, a computer program has been written that takes every individual's response on how the automated system would affect the time they spend in each functional area, i.e. whether it would increase or decrease and by how much, and applies it to what they said was the relative time spent in that area now. For example, if a supervisor/trainer said he/she would spend 40% less time documenting training, and they answered earlier that they spend 25% of their current training time documenting, the computer would decrease the 25% block of training time by 40%. Again as an example, if the total of the above calculations came to 90% and the supervisor/trainer said early in the survey that they spend about 40% of their duty time on training-related activities, then under an automated system, they might spend only 36% of their time on training (.90 x 40). To derive the dollar value of this change, one merely needs to calculate what 4% (40 - 36) equates to for the respondent's grade and time in service.

SUMMARY

This report has presented in some detail the characteristics and capabilities of the AOTS Cost Model. The model can provide estimates of total LCC incurred by the AF as a result of incorporating various AOTS concepts into the training process. Furthermore these costs are categorized via a detailed WBS, which allows individual cost drivers to be identified. Hardware costs attributable to 2-digit AFSs and MAJCOMs can be broken out in a similar fashion. Once an AOTS configuration has been selected for POM analysis, data from the primary AOTS model is used to spread costs over a specified number of years. The POM model then provides a means of supplying cost estimates for AF budgeting purposes.

The report described an approach to assessing the benefits that could be expected with an automated on-the-job training system such as AOTS. Although data analysis is incomplete, a brief description of questionnaires and interview techniques administered to representative Air Force enlisted members was described.

WHERE DOES CBT FIT IN, NOW THAT WE KNOW SO MUCH?: A FRONT END ANALYSIS STUDY

Andrew E. Andrews and Mary Stoddard Trainor
Cognitive Engineering Design and Research Team (CEDAR)
Military Systems Group
Mail Stop F601, A-6
Los Alamos National Laboratory
Los Alamos, New Mexico 87545

ABSTRACT

Computer-based training (CBT) has now been in existence for over two decades. It has been implemented in both the private sector and government organizations at an exponential rate. Nevertheless, many institutions, particularly educational institutions, have not yet introduced CBT. Our knowledge of what works and what does not, as well as hardware and software advances, has greatly increased in the past few years. This paper addresses many management considerations with respect to CBT. First, we consider the generic environment in which CBT might be used and then issues that affect costs and benefits, including lessons learned by the Cognitive Engineering Design and Research Team (CEDAR) of the Los Alamos National Laboratory in its assessments. The final section gives some "how-to" guidelines on increasing the probability of successfully introducing CBT into the training environment. The underlying theme of the paper is that management should be guided by what we now know about costs and benefits in its decisions regarding CBT and fight the lure of "high tech" glitter.

INTRODUCTION

Since computer-based training (CBT) has been in existence, we have seen the field progress from using the computer as a control for electronic page turning to the current state-of-the-art systems that permit a wide variety of instructional strategies. Additionally, we have the expectations that computers can now think like instructors and thereby dialog with the student.

If one peruses any recent issue of the popular computer magazines dealing with microcomputers, one can find several advertisements offering rather complete systems for less than one thousand dollars. If one looks at colleges or universities, such as Stanford or Drexel, the use of computers to support the curricula is readily apparent. At Drexel, all students must have access to a personal computer and use them in all courses throughout their four years of college.[1] If one looks at the CBT literature, one sees many studies touting CBT as the answer to such instructional problems as self-pacing, reaching the advanced student, laboratory or simulation shortage, and preserving instructor time. So why shouldn't any institution wanting to use modern technology, reduce costs, and implement a CBT program?

The answer is that this simple, casual promise of CBT is not simple and cheap, or necessarily the best course of action for the institution. In fact, a recent Army Research Institute report asserts that clear-cut benefits of CBT have not been demonstrated.[2]

This paper deals with why one should choose to adopt a CBT program and, assuming a positive choice, some guidelines on how to go about it. The bold assumption is that to see a definite advantage or benefit of CBT commensurate with its cost, great care must be exercised in the selection of applications and in justifying CBT based upon its merits alone. Many of the questions that should be asked during the front end analysis process are identified.

ENVIRONMENTAL IMPACT

The impact of CBT is dependent upon the environment in which it is used. As a simplistic illustration, one would not place a CBT unit at a swimming pool to teach Olympic hopefuls better butterfly stroke technique. The presence of water is an essential element that cannot be simulated by the computer. In contrast, welding has been effectively taught using CBT, with emphasis on simulating the welding process.[3] On a general level, CBT can be used in three different environments:

CLASSROOM: A formal training environment in which performance can be measured in terms of terminal performance objectives. CBT in this environment can be used either as a substitute for classroom or laboratory instruction or as a supplement to conventional instruction. Often, because of differing physical needs for CBT, a separate learning center used by several different classes is built and monitored by advanced students or by support personnel. The specific strategy of the CBT depends upon how the lessons are implemented relative to the classroom.

ON-THE-JOB: A less formal environment in which improvement is more difficult to measure. Generally, the "instructor" is the front-line supervisor whose principal job is other than training. CBT offers the opportunity for standardization of instruction as well as improved quality, but its effectiveness is difficult to measure. This category subsumes many subcategories including apprenticeship, sustainment, and retraining for new equipment.

EXTENSION COURSE: Usually not required of the employee, but made available on a basis similar to "continuing education." Benefits to the "company" are extremely difficult to measure and such programs are supported on the premise that better educated employees are better employees.

*This work was partially supported by the Army Research Institute.

Within the Department of Defense, however, extension courses are essential to accomplishing the training mission and may be required.

The importance of distinguishing among these environments is that CBT, which may have great benefit in one environment, may be of little value in another. For example, standardization may be of great importance for the National Guard with respect to instruction that must be exported to the field (extension courses). Consistency in quality of instruction may be essential if the largest number of trainees is to achieve a minimum acceptable level of proficiency. In contrast, for the active forces consistency is important, but quality controls on instructor presentation are inherent in the classroom environment. Hence, an advantage of CBT in one environment (the National Guard) may not be a worthwhile benefit in another (the Active Components).

COST AND BENEFITS

There are several reasons for introducing CBT into the training environment, including the following:

- Improving the cost/benefit ratio with respect to the training of personnel. Cost is the total expense (both fixed and variable) associated with training an individual. Benefit reflects the difference between the value of the trainee to the organization before and after training. The goal is the lowest cost/benefit ratio possible (note that costs and benefits are always positive values that can approach but not equal zero).

- Providing training that is otherwise not feasible (for example, extension courses).

- Doing research into CBT. Academic departments, industry, and organizations, such as CEDAR, engage in this type of activity. The benefit is knowledge gained on how to do CBT better (or perhaps what to avoid).

- Improving the image of the organization. Image is an elusive quality and its importance should not be overlooked. It is similar to "goodwill" that is paid for when a company is purchased. As such, it is a benefit assessed only in subjective terms.

- Making a capricious decision by management to do it. Management may decree that CBT will be used without providing the rationale to the organization. Of course, it may not be capricious. Rather, management may know what it wants to do but, as is the case with many experts, cannot or does not believe there is a need to explain the rationale.

While the last three reasons can have great merit in certain circumstances, they do not withstand hard-nosed management examination in the context of profit and loss. Instead, a decision in favor of CBT should be based on an improved cost/benefit ratio. That is, can costs be reduced, benefits (as seen in better trained people) be improved, or both?

Benefits of training should be measured in terms of the organization. From a business perspective, one trains people to increase productivity. And, if people with the requisite skills can be hired directly without additional cost, this choice is the preferred one. This approach has severe limitations, however, because a person's heuristic knowledge base is developed on-the-job. In many businesses, particularly national defense, the requisite skills are not taught elsewhere.

A good way to assess the benefit of CBT is to introduce it on a small scale and measure its effectiveness in a controlled manner. However, success is directly related to the quality of the implementation and does not necessarily indicate future success of a broader scale implementation. In the business sense, one would like to forecast the gains of CBT, that is, make an estimate of the near-term benefits based on some sort of regression analysis from past results. Generally, the benefits of CBT can be predicted based on experience in the field and the application of heuristics derived from it. Quantified predictions of benefits should be viewed with great skepticism.

Doing a cost/benefit analysis is a difficult task at best. And when management is considering a new field or application, the complexity of the field can obfuscate otherwise obvious factors from consideration. In this section, a few critical factors that should affect a decision for or against implementing a CBT effort will be discussed. The discussion will lead to identifying these CBT applications with the greatest potential return on investment.

As already observed, the benefits to be derived from a new CBT application must be predicted, not forecasted. As such, benefits (before the fact) represent sophisticated hand waving and (after the fact) frequently correlate to the quality of the implementation. The quality of courseware design largely determines the success of CBT. Many comparative studies have been performed comparing CBT to conventional instructional methods.[5] These studies show that CBT can be more effective than conventional instruction, but the degree of effectiveness (and hence benefit) depends upon design issues as well as the local situation.[6]

Up front, CBT usually represents a more costly approach because of the high initial investment.[7] The low priced computer systems lend themselves to the old electronic page turning techniques but do not necessarily support modern instructional technology. The instructional strategies of simulation and gaming, among others, require more sophisticated technologies. Reaping the benefits of CBT for your application might require a spectrum of capabilities that can include interactive video disc, digital audio, graphics, color, data and program storage, compact disc read-only-memory, computational speed, multiple displays, and simulation. The list can go on and is limited only by one's imagination. Yet, central to the list are both the cost of acqusition and the cost of courseware to be run on the system.

In general, the cost of courseware development will greatly exceed the cost of equipment.[7] Equipment acquired today probably will be obsolete five years from now. It is therefore necessary to be requirements-, not technology-, driven.

In estimating the cost of CBT, the price of equipment and facilities usually can be established in a fairly sound fashion. The cost of development of courseware, maintenance, administration of the program, and the time employees devote to learning can be only imprecisely estimated at best. These factors are interdependent and nearly impossible to predict for creative endeavors.

Nevertheless, it is clear that CBT cannot replace instructors, only free them up to spend their time aiding individuals and in lesson design, production, and maintenance. The roles of instructors will change, but the manpower commitment will remain and may grow. Of course, classroom instructors may not have the skills for CBT development.

Table I, for example, lists the talents required to develop and produce good quality CBT using interactive video. The breadth of skills required leads to an argument against the assertion that CBT cannot replace instructors. If courseware is to be contracted, perhaps the size of the training department can be reduced. Further, the courseware company can take the lessons already taught; put them on a computer; and, hence, eliminate the need for lesson design, development, and maintenance. The fallacy of this argument has two aspects. First, contracting for courseware production does not eliminate the in-house manpower costs for courseware development but shifts them (perhaps increases them) to a different line item. The second aspect is that CBT, which consists of straight conversion of a classroom course, is generally not successful. Revision of the instructional design is required. The implication is that CBT is going to cost more than classroom instruction.

TABLE I

REQUIRED TEAM SKILLS

- Subject Matter Expertise
- Computer Science
- Cognitive Science
- Human Factors
- Instructional Design
- Graphic Arts
- Script Writing
- Video Expertise
- Management

AND A GOOD WORKING ENVIRONMENT

Now wait a minute! If the benefits of CBT are hard to predict (often being sophisticated hand waving) and costs are likely to go up, why do it? The answer lies in the potential of CBT benefits, that is, what CBT can do that conventional training cannot and what CBT can do better than conventional training. The point is that CBT represents a risk with significant rewards for the innovative, aggressive training program.

WHERE SHOULD CBT BE USED?

The key to success is in selecting appropriate applications for CBT--those that cannot be achieved by other means or those in which a moderate CBT investment can provide other savings. For example, a CBT simulator could serve as a part-task trainer to teach "switchology," thus taking the training burden from more costly simulators.[9] Selection of CBT implementations should be based on what CBT can do well as evidenced by improved performance or permitting achievement of a teaching strategy not easily achieved through other means.

Looking at Bloom's Taxonomy (Table II), most training today is at the lower cognitive levels. Yet, there is a growing awareness of the necessity to provide good training at higher cognitive levels. Students need to go beyond the facts and procedures of the classroom and experience real world dilemmas. In essence, it is desirable to give the student artificial experience before he tries it in actuality, thus improving his chances of good performance. CBT can be used for high cognitive level objectives (for example, synthesis or analysis), but the design time required is greater than for lower level objectives (for example, comprehension and knowledge) because the instructional strategies are more complex (for example, simulation and gaming).

TABLE II

BLDDM'S TAXONOMY

(high cognitive level)	• Evaluation
	• Synthesis
	• Analysis
	• Application
	• Comprehension
(low cognitive level)	• Knowledge (recall)

(Adapted from: TAXONOMY OF EDUCATIONAL OBJECTIVES: The Classification of Educational Goals: HANDBOOK I: Cognitive Domain, by Benjamin Bloom, et al, (Longman, Inc., 1956).

Simulation means different things in different contexts. With respect to the training environment the term can include physical, procedural, situational, and process simulations.[10] The differences between games and simulations are twofold. First, games require competition, either with the computer or with another player. Second, games focus on broad, less quantifiable concepts (soft concepts), while simulations are concerned with highly accurate, technical detail (hard concepts). Simulations are required to correctly predict a great many details, while games are not. A comparative matrix is shown in Table III.

TABLE III

A GAMING VERSUS SIMULATION MATRIX

	Gaming	Simulation
Purpose	concepts	analysis
Prerequisites	fewer	more
Need to understand "the system"	yes, but can learn while playing	yes
Fidelity	not as critical	must be high

The distinction between games and simulations is critical with regard to the development effort. If you require a simulation when a game would suffice, you will spend more money than is necessary. Also, if you do not have an instructional strategy in mind, both games and simulations may be the wrong choice. The use of computers for educational purposes without a strong, underlying instructional strategy that matches human need will produce suboptimal results.

As a bottom line of cost/benefit, CBT has certain applications that make it an attractive alternative and worthy of careful consideration. These applications are as follows:

- Simulation of equipment to support procedural training.
- Gaming and simulation to support the acquisition of artificial experience.
- The export of training (at all cognitive levels) to make it more widely available and consistently good.

GETTING INTO CBT

At some point, you get a visceral gut feeling that CBT is required. You see some potential applications, and the other alternatives are not as attractive. You have made a rough-cut estimate and believe that the potential rewards justify the risk. How do you go about it such that a high probability success path is followed? Table IV contains some guidelines that are discussed below.

TABLE IV

GUIOELINES FOR THE INTROOUCTION OF COMPUTER-BASEO TRAINING

o Allow time for a front end analysis to determine if you have a training problem or a performance problem.

o Obtain support from high-level management early in the process and then make an effort to continuously foster it.

o Determine who is in charge--establish a focal point for CBT.

o Assemble a diverse development team.

o Establish the training requirements, enumerate potential applications, prioritize, and select the one with the greatest possible payoff commensurate with acceptable risk.

o Involve instructors in the design process and ensure that they are adequately trained regarding the CBT medium.

o Gradually introduce the new training approach. Let the instructors and students become accustomed to it and then become the prime advocates.

o CONTINUALLY REVIEW THE COSTS ANO POTENTIAL BENEFITS OF YOUR CBT PROGRAM ANO OEMANO THAT CBT BE COST EFFECTIVE OVER OTHER MEANS.

First and foremost, allow time for a front end analysis to determine if you have a training problem or a performance problem. If the worker has the knowledge, skills, and abilities required for the task, you probably do not have a training problem. Often, the true problem may be obscured by the organizational environment. For example, operational policies and procedures may be inhibiting creativity and initiative on the part of the worker, thus ensuring continual inefficiency.

The second step is to obtain support from high-level management early in the process and then make an effort to continuously foster it. This support is essential to success. The initial investment for CBT equipment is too large to obscure within the budget. However, on a continuing basis, CBT will have to fight with other budget items until it is established, a process that could take several years.

Next, determine who is in charge--establish a focal point for CBT. In organizations we have visited and observed, those that did not follow this guideline tended to have a variety of equipment and multiple standards for CBT quality, and lacked flexibility with regard to the exchange of materials. Without a single point of contact, a CBT program can quickly look like the start of a computer thrift shop. At the same time, the people on the implementation team must recognize that centralization benefits them and that they can get the resources they need as long as they are responsibly flexible regarding certain details. The focal point of the CBT activity must be sensitive to corporate needs, operational constraints, the operative technologies, and both the implementers and users of the training system. Conflicts among these variables will occur; the focal point for CBT is the focus of conflict resolution and the link to continuing management support.

CBT is a team effort that requires the skills shown in Table I, or a variant of it. The next step is to assemble a diverse development team or select a contractor with one. Assembling the team yourself requires a commitment to team building. For example, script writers and computer programmers view the world differently and have different requirements to accomplish their jobs. Yet, to be successful, a CBT team must communicate within itself, and the members must adapt to one another. A separation of functions leads to lower quality, less creative CBT. By implication, CBT lends itself to project management techniques and a matrix management approach. However, if you cannot assemble a team with all the requisite skills, look for help elsewhere.

With the team assembled, revisit the training requirements, enumerate potential applications, prioritize, and select the application with the greatest possible payoff commensurate with acceptable risk. Note that to this point no mention of hardware acquisition has been made because you should be needs-driven, not technology-driven. Choose equipment that will support your priority courseware requirements but has the potential for expansion to support all the courseware requirements. For example, if you need to teach switchology, you almost certainly will need a good graphics capability but may not require interactive video disc, thus reducing capital outlays while you are on the steep part of the learning curve. Also, opt for applications that CBT can do well. If you have a choice between teaching workers the steps in a process by rote memory or how to set up equipment through a procedural simulation, opt for the latter because it matches what CBT can do well while having a good potential return on investment.

Keeping costs down also helps with winning and maintaining upper management support. First, by purchasing only the hardware capabilities required, costs are minimized. Second, by focusing on the courseware with the highest priority and best payoff, you optimize the potential benefit and produce recognizable results in minimal time. The cost/benefit ratio will be clear, near-term evidence of upper management's wisdom in supporting CBT.

Next consider what you may be doing with regard to the existing training organization. At the very least, the introduction of CBT represents change. At the other end of the spectrum, CBT threatens the jobs of the instructors. The existing training team will resist the introduction of CBT unless they are participants in it. However, simply being asked or directed to participate does not mean the problem is solved. The trainers also must understand what CBT is about and how to do it. Be prepared to train the trainers. This point can be stated as the following: Involve instructors in the design process and ensure that they are adequately trained regarding the CBT medium.

Just as CBT causes change in the instructor's environment, it causes change in the student's world. To be successful, the inertia of the traditional learning experience must be overcome. While at some time in the future the population will regard computers in the classroom as commonplace, the vast majority of today's work force experienced a more traditional approach to learning during their formal schooling.

Gradually introduce the new training approach. Let the instructors and students become accustomed to it and then become the prime advocates. In essence, let both student and instructor, by themselves, evaluate the evidence of student performance both with and without CBT. A corollary implication is that the courseware for application selected for the introduction process should support the self-evaluation process. For example, a CBT-type part-task trainer can help students perform with greater skill and confidence when they advance to full system simulators.

WELL, THERE YOU HAVE IT!

A look at the costs and benefits of CBT, what CBT can do best, and some guidelines on how to do it. For convenience, the guidelines are gathered together in Table IV. With these guidelines and the lessons listed earlier, is there a central theme or single, pervasive guideline that should be followed? Yes there is!

> CONTINUALLY REVIEW THE COSTS AND POTENTIAL BENEFITS OF YOUR CBT PROGRAM AND DEMAND THAT CBT BE COST EFFECTIVE OVER OTHER MEANS.

The cost/benefit ratio for the CBT solution must be better than the other potential solutions. While the decision criterion is simply stated, getting to the decision point is a very complex issue. There are many underlying considerations that include who, what, when, where, why, and how. CBT represents a risk or gamble. And while CBT may be akin to the glitter and glamour of gambling in Las Vegas or Atlantic City, winning likewise demands concentration on the fundamentals--here, teaching and learning. If you avoid the lure of high technology and demand a solid, comparative, decision base, use of CBT when supported by the evidence will result in better training.

REFERENCES

1. Visit by Charles Thorn, staff member CEOAR, to Orexel University, April 14, 1987.

2. Shlecter, T. E., "An Examination of the Research Evidence for Computer-Based Instruction in Military Training," ARI Technical Report 722, August 1986.

3. Demonstration by David Hon, president IXION, Inc., at the TITE Conference, Colorado Springs, CO, March 16-20, 1987, and reported in Memories Optique, No. 33, November 1986.

4. Shlecter (op. cit.) reports that the transportability of research findings across disciplines, different instructional contexts, and media are problematic because of the high degree of impact these variables may have had on the results.

5. Deigran, C. B., et al, "Computer-Assisted, Programmed Text, and Lecture Modes of Instruction in Three Medical Training Cases: Comparative Evaluation," Air Force Systems Command Report AFHRL-TR-79-76, July 1980.

6. Kulik, J. A. and Jaksa, P., "PSI and Other Educational Technologies in College Teaching," Educational Technology, 12-19 (September 1977).

7. Levin, H. M., "Costs and Benefits of Computer-Assisted Instruction," Public Budgeting and Finances, (Spring 1975).

8. Held, T., Reeves, T., and Aggen, W., "Designing and Producing Computer-Based Interactive Video Courseware," Notes from a Tutorial Presented at AOCIS, New Orleans, LA, February 1985.

9. Roman, J. H., Pistone, R. A., and Stoddard, M. L., "A Trilevel Interaction Oesign Model for Pilot Part-Task Training," in "Proceedings of the Human Factors Society 30th Annual Meeting," 1986, pp. 126-130.

10. Alessi, S. M. and Trollip, S. R., Computer-Based Instruction: Methods and Development (Prentice-Hall, New Jersey, 1985).

11. Crawford, Chris, Balance of Power (Microsoft Press, Washington, 1986) pp. 6-11.

ABOUT THE AUTHORS

ANDREW E. ANDREWS heads CEOAR Team, Military Systems Group, Los Alamos National Laboratory. The CEDAR Team has pioneered the concept of the Computer Tutor, a bridge between the computer-based training and the artificial intelligence communities. Andrews holds an M.S. in Nuclear Engineering from the University of California at Berkeley, an M.B.A. from Long Island University, and a Master of Military Arts and Sciences degree. Prior to his work in CEOAR, he held varied positions in the U.S. Army that included exercise and training development. Additionally, he served as a military research associate at the Los Alamos National Laboratory and culminated his Army career as the senior air defense staff officer at the Supreme Headquarters Allied Powers Europe. Andrews has several publications in engineering, military doctrine, and training technology.

MARY STODDARD TRAINOR is a technical staff member of the Cognitive Engineering Design and Research Team (CEDAR), Military Systems Group, Los Alamos National Laboratory. She provides expertise in instructional design, user interface design, computer-based training, and cognitive science for CEDAR. She has been a principal contributor to the development of the computer tutor concept. Trainor has been at Los Alamos for six years, with prior positions in the computer user education, documentation, and information systems areas. She holds a Ph.D. from the Group in Science and Mathematics Education at the University of California at Berkeley. Trainor has 25 publications, with her most recent in the areas of intelligent tutoring systems, computer tutors, and human factors in computing.

MANAGING TRAINING DEVELOPMENT AS A MANPRINT ELEMENT

Grace P. Waldrop
HAY Systems, Inc.
Training Systems Division
12424 Research Parkway, Suite 250
Orlando, FL 32826

ABSTRACT

The Manpower and Personnel Integration (MANPRINT) initiative has generated numerous concerns and issues in all of the affected disciplines (human factors engineering, manpower, personnel, training, health hazard assessment, and system safety). Unique technical approaches must be developed to integrate the multi-disciplinary data elements within traditional analytical procedures. Equally important is the management of the initiative, between and within those elements. Training and training devices (training systems), while an integral part of the MANPRINT program for the weapon system development, necessitate special consideration as a "system within a system." This consideration requires industry to carefully plan, manage, and integrate their approach to training system design as part of the overall weapon system design. This paper will address the issues surrounding organization, planning, and management of training system development as a MANPRINT element. Unique management approaches and examples will be provided rather than a technical review of training development.

INTRODUCTION

The U.S. Army's Manpower and Personnel Integration (MANPRINT) initiative focuses on early design impact of six key disciplines: Manpower, Personnel, Training, Human Factors Engineering, Health Hazard Assessment, and System Safety. Each of these disciplines has a critical role to play within the weapon system development process. The role requires an interdependent approach to coordinate technical data exchange and utilization. Training programs and training devices (training systems) are an integral part of the MANPRINT initiative and should be considered as a "system within a system." This consideration leads to a complete MANPRINT cycle within the discipline of training, i.e., all six MANPRINT domains are embedded within training system development. While the issues and concerns surrounding MANPRINT affect both industry and government, the primary perspective here is from the industry viewpoint.

The MANPRINT initiative was started to optimize total system performance. In an effort to achieve performance goals we have attempted to utilize the analytical methods which strive to optimize resources, (people and equipment), and thus to minimize performance problems resulting from a lack of optimization. MANPRINT methods or analytical techniques, arguably not new, necessitate new approaches to data utilization and creative tools for data application and analysis. To be effective within the concept of MANPRINT, management of training systems development includes utilization of appropriate weapon system data; integrated through the MANPRINT management to optimize performance and resources. These data include manpower issues (personnel quantity), personnel factors (personnel quality), human factors data (interface and tasks), safety and health issues (situations which necessitate specialized training or design). The MANPRINT initiative can serve as the vehicle to coordinate these types of data, beginning at the very front-end of the design process. Traditionally training system design, while "in the program structure," has not been actively involved with these design disciplines or the system's design engineers. Learning to utilize these data to the program's advantage (a total system perspective) is one of the training system developer's MANPRINT challenges.

The process, however, does not end with integration of training into weapon system development. The MANPRINT challenge is part of the training system development process itself. MANPRINT related issues for training system development might include: the number of instructors available (manpower), the knowledge level of available instructors (personnel), trainer fidelity and interface criteria (human factors engineering), equipment characteristics with potential danger (system safety), and training required to fire the weapon (health hazards assessments). Certain of these issues are common to any training development process and thus do not require a repetitive technical discussion here. However, and somewhat previously neglected, is the management portion of a MANPRINT program. It can provide a vehicle to track and document program material and then insure that appropriate technical elements share and utilize common data.

Training systems, like their counterpart weapon systems, range from small to quite large and cover a wide variety of technologies. Regardless of the type, the training developer cannot work in a vacuum to create instructional materials or create a training device. The developers (sponsors and contractors) will have to more proactively participate in the design process and more

aggressively solicit appropriate data. A well-versed technical staff will be ineffective without a management forum interacting with design engineers, such as that developing in response to the MANPRINT initiative.

Unfortunately, it is widely recognized that until MANPRINT related concerns can be funded as line items and reasonably measured in terms of trade-offs it will continue to have less than full corporate support. This is not surprising due to the fact that all six MANPRINT domains have always suffered the same lack of support for essentially the same reasons. Both government and industry are to blame, in part, due to a lack of understanding and an emphasis on hardware engineering. Government is to blame because it has not required that these analytical techniques be utilized. Industry is also to blame because its only interest is what government is willing to pay for. MANPRINT programs, if setup properly and managed efficiently, can serve to impact system designs with measurable soldier related data, which, have previously never been fully utilized. The MANPRINT process can be priced and produces measurable results (e.g., quantitative manpower requirements analysis). What has been missing is technical data utilization coupled with a defined management approach to support the techniques which impact resource optimization.

CHANGING ROLES

Traditionally, the roles of the industrial training community have been pretty much confined to positions outside the mainstream of the acquisition process. Input into the mainstream is usually in the area of program status updates. The MANPRINT initiative has provided an opportunity for the training community to break with tradition and become an integral part of the overall process. However, this opportunity also requires that the training community utilize data provided through the MANPRINT program and to provide timely feedback to the sponsor (to give to the Program Manager) if any program resource constraints and goals cannot be fully complied with.

The traditional training development strategy is driven by requirements imposed by the overall acquisition strategy. This role easily results in a one-way flow of data, i.e., from the overall program to training. The traditional role of data receiver should give way to more complete training data integration. Integration includes accepting data on the number and types of personnel involved (operators, maintainers, students, instructors), design considerations which impact device development, etc. and utilizing it to the program's advantage by capitalizing on available resources. The integration equation also includes providing data (training pipeline options, student/instructor ratio tradeoffs, etc.) to the user which is necessary to make informed choices. The changing role, from "data receiver to data user," is not without burdens. These include the burden of verification of target audience appropriateness, meeting manpower and personnel constraints, and insuring acceptable and safe designs.

Another aspect of MANPRINT which leads to a change for some developers is the involvement of the user. The government and industry training groups cannot validate compliance to resource constraints without close cooperation with the ultimate user. The data provided by this user help to insure that performance of the system meets user requirements for its intended environment. This user is many times is also a valuable source of background information, predecessor system data, and documents source. The MANPRINT initiative seeks to eliminate parochial generation of new data, which might already exist in some form (probably from the weapon system or predecessor training system).

Two roles for training personnel evolve within many programs; i.e., examining training requirements relative to weapon system fielding and developing training programs for the training device. Unfortunately, these roles are often performed in different programs (fielding the weapon system or fielding the training devices) and have little or no interaction. The management structure must provide non-constraining methodologies to provide access between groups. Training personnel require technical input from all six domains as opposed to interaction as one of six domains.

MANAGEMENT ISSUES

While the ultimate goal of MANPRINT is to technically impact the system's design with soldier considerations, the management techniques used can support or hinder achievement of any design impact. Industrial organizations are no better equipped than the Army to influence system design, early-on, with soldier resource considerations. However, creative management techniques can prevent many technical obstacles and increase MANPRINT related impacts. The first management consideration is where to organizationally place the MANPRINT program. The Army has directed that its internal program be within the Integrated Logistics Support (ILS) group and many industry primes have followed suit. The effectiveness of this approach is company specific and dependent upon where the individual MANPRINT domains are organizationally placed.

If the MANPRINT program is located in the ILS group but several of the functional disciplines are in systems engineering, extra caution must be taken to insure the open communication so critical to a MANPRINT program. Corporate management rarely realizes the necessity to provide MANPRINT disciplines a working structure to cross divisional

lines on an informal and formal level (Figure 1). The informal provides a daily technical information exchange for sharing analysis data. The formal structure provides a vehicle to support deliverable requirements and staffing of MANPRINT efforts. The formal structure also provides a coordinating management organization to support requirements outside of the core MANPRINT efforts, e.g., when data are required from ILS personnel. Industry has not found it advantageous to have personnel in each MANPRINT domain asking for a variety of data elements, but it is effective to have a centralized coordinator (MANPRINT Coordinator or Manager) for these activities.

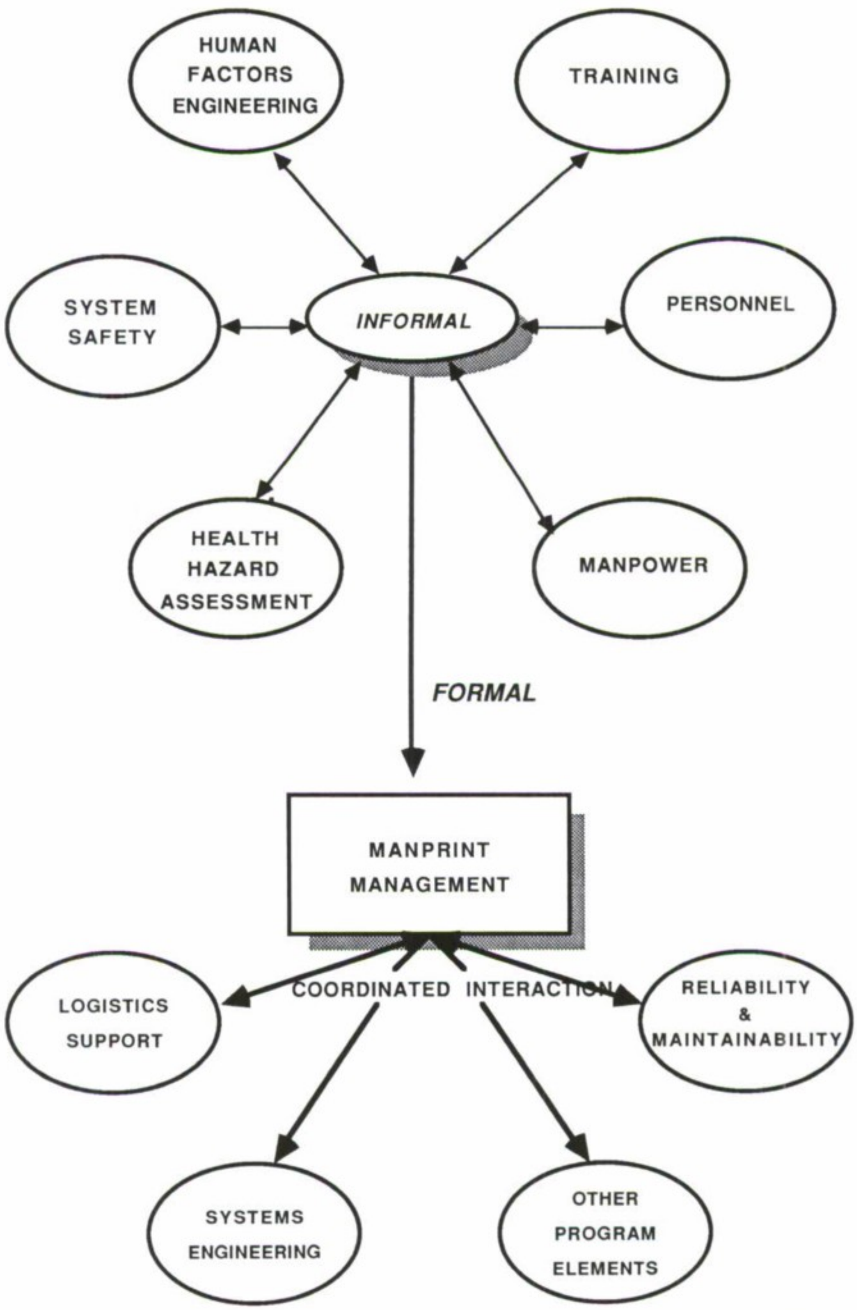


FIGURE 1. EFFECTIVE MANPRINT PROGRAM MANAGEMENT IS BASED UPON FORMAL AND INFORMAL INTERACTIONS

The most common approach thus far within industry is to task the ILS Manager with MANPRINT duties and name the position "ILS/MANPRINT Manager." If this is the only change and concession, it is not indicative of a strong MANPRINT commitment since the ILS Manager presumably already had full-time duties. Problems with this approach are compounded if some of the MANPRINT domains are outside the ILS Manager's line-of-authority. In large scale programs, this approach would be unworkable without a dedicated MANPRINT Manager who would use the ILS/MANPRINT Manager as "higher authority" for program level decisions. The same is true for those companies who create a System Engineering/MANPRINT Manager.

The training domain is particularly sensitive to the effectiveness of the organizational structure since it requires input from and interaction with all domains. Therefore, training must interact within the MANPRINT program both at a management level and at a detailed technical level to complete the analytical requirements (Figure 2). This level of MANPRINT interaction requires a MANPRINT program internal to the training systems development structure (i.e. building a training device).

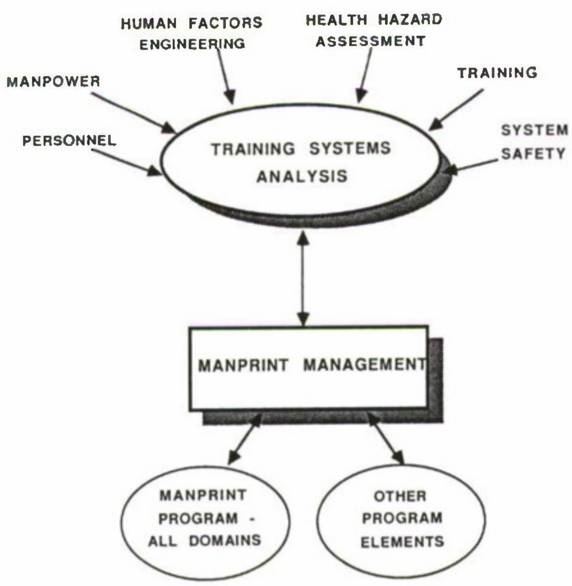


FIGURE 2. TRAINING SYSTEM ANALYSIS REQUIRES TECHNICAL INPUT FROM ALL MANPRINT DOMAINS AND MUST INTERACT WITH MANPRINT MANAGEMENT AS A DOMAIN

INTER-TRAINING MANPRINT PROGRAM

Once a weapon system has been identified for development, off-the-shelf acquisition, etc., to include a MANPRINT element; industry activities (to include MANPRINT) should begin. The same is true for training procurements involving training devices. To incorporate the system design within MANPRINT constraints, the training group must actively and early on participate in the generation and utilization of resource-related data. Management of this effort is illustrated in Figure 3. User data are major common components in both MANPRINT efforts represented in this Figure.

Critical to making this model work is development of a program management approach to include MANPRINT in the milestone schedule. This is critical because if the training milestones are keyed off the weapon system requirements then the weapon system MANPRINT milestones can drive the training MANPRINT deliverables. This level of detail is important because it emphasizes to all program participants the criticality of putting MANPRINT early in the process and the inter-relationship of MANPRINT and the more traditional program elements. Without becoming an integral part of the program schedule, MANPRINT will never become an important program element (in training or weapon system procurements).

Realistically training, particularly a major device or simulation, is procured separately from the weapon system. This adds an extra burden, i.e., coordinating technical and management issues between completely separate programs. Procurement cycles rarely track cleanly and data are not exchanged easily unless a teaming agreement between companies incorporates training or unless the data are provided as Government furnished information. The worst case (and all too common) scenario is one where the training device procurement agency is not itself coordinated with the weapon system's program manager. This can easily result in a lack of timely data and creation of redundant data. Both of these results are all too common in training system development efforts. While data coordination has always been a program issue, MANPRINT may provide the vehicle to actually begin to address the data problems. The initiative is forcing both industry and government analysts to examine data availability and its application.

Redundant Data

MANPRINT efforts suffer the compounded effects of six domains which are not readily assimilated into the traditional engineering data structure. For example, while all MANPRINT domains can and should benefit from the integrated logistic support (ILS) database, it is not required and not widely understood within many of the

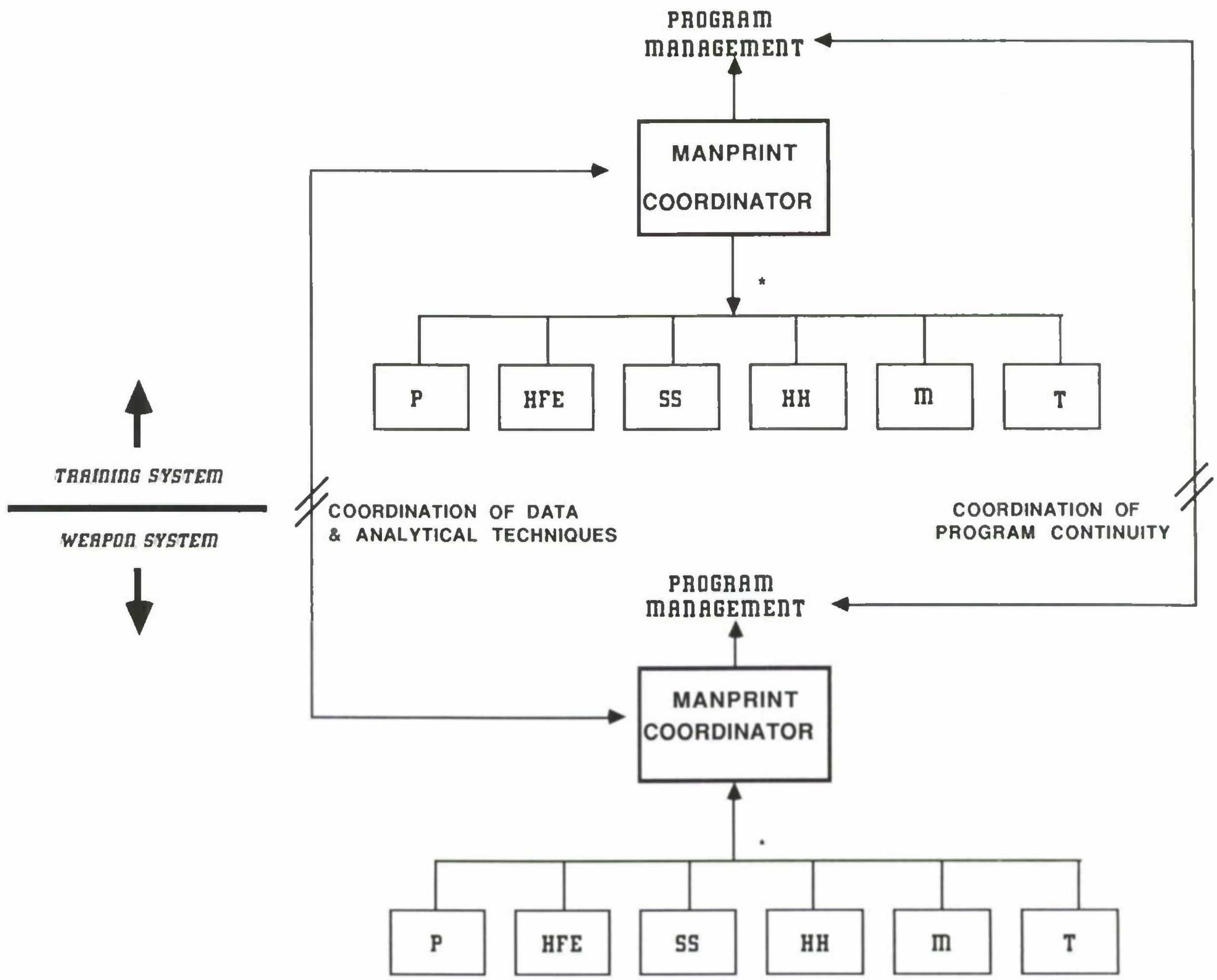


FIGURE 3. IDEALLY, TRAINING DEVICE PROCUREMENT SHOULD HAVE AN INTERNAL MANPRINT PROGRAM AND PLANNED COORDINATION WITH THE WEAPON SYSTEM MANPRINT PROGRAM

***Domain personnel may work both efforts.**

MANPRINT technical disciplines. Similarly, the ILS personnel rarely actively seek input from the MANPRINT-related disciplines. The resulting base of information is hardly consolidated.

The easiest example of redundant data development is the task analysis. Programs usually have several versions of task analyses, created for slightly differing results. Whether function or mission oriented, task analysis data is a primary candidate for data sharing. A consolidated data base is important to all MANPRINT activities and particularly critical to development of a timely training system since the task analysis feeds training media decisions and training documentation.

Lack of Timely Data

Training development lags behind the weapon system for obvious reasons; i.e., we have to know what the training is being developed to support. However, today's sophisticated weapon systems are requiring more complex training systems and longer pipelines. If training is to meet fielding requirements for many systems, it must be developed very early in the acquisition process.

The type (e.g., predecessor), amount (e.g., operator only), and level of data detail (e.g., high order) become increasingly important as training development is moved "to the left" in the acquisition process. Like any good training program, MANPRINT will rely upon predecessor systems and lessons learned, to form a baseline for analytical processes to meet deadlines. The MANPRINT Manager for the weapon system can provide the data from analytical techniques and the training MANPRINT Manager should disseminate these data throughout the training development process. Training analyses benefit from the same raw data (predecessor or system specific) utilized in the weapons system development, e.g., logistic support analysis records, reliability and maintainability reports.

When the program management plan is developed to include MANPRINT, the time-lines must consider data input requirements. The MANPRINT program

milestones are those embedded in each domain, e.g., Human Engineering Design Approach Documents; with the addition of MANPRINT Working Group Meetings and other management events. These types of milestones and deliverables must be utilized and required as part of the milestone schedule and deliverables for the more "main stream" engineering disciplines before any MANPRINT elements are on the critical path.

The MANPRINT domains embedded within the training program must develop a data utilization structure, based upon the level of specificity available. For example the human factors engineer cannot wait until the first training device rolls off the line to make design recommendations. Rather, he or she must utilize drawings, ILS task data, subject matter expertise, mockups, etc., to iteratively analyze requirements. The training system development process, like the weapon system development process, can only benefit from MANPRINT if it is conducted early and continued throughout the process with timely results.

SUMMARY

Training system development maintains a unique position within the MANPRINT initiative. First as a MANPRINT domain, training must fit within specified resource constraints. Training pipelines can no longer be expected to increase arbitrarily or to "fix" design problems. Second, as a device builder, training programs must integrate and utilize all the MANPRINT domains. Training devices must be designed with considerations forresource utilization and interface similar to the weapon system.

Management of these efforts can be as critical as their analytical techniques in terms of a successful MANPRINT program. The MANPRINT initiative is not necessarily imposing new analytical requirements, but requires us to use and apply those we already have available and to develop new tools for making them work. The management tools which can make this happen include consolidated database creation and utilization of both formal and informal data integration structures. The management structure must result in a fully integrated design approach which not only integrates the MANPRINT domains but incorporates <u>all</u> design disciplines.

ABOUT THE AUTHOR

Grace Waldrop is Program Director for Human Factors Engineering with HAY Systems, Inc. She holds a Master's degree from the University of Central Florida in Industrial Psychology. She has 10 years experience in human factors and training device developments. Ms. Waldrop most recently has participated in technical and management tasks on programs with MANPRINT elements for both weapon systems and training devices. She has also taught the HFE section of the Army's MANPRINT training and has conducted industry MANPRINT seminars. HAY has been instrumental in development of technical and management tools to facilitate implementation of MANPRINT.

ARMY COMBAT TRAININC CENTERS TEN-YEAR VISION

COLONEL WALTER C. ZAREMBA
US ARMY TRAININC SUPPORT CENTER
FORT EUSTIS, VIRCINIA 23604-5166

ABSTRACT

The Army's Combat Training Centers (CTC) include the National Training Center (NTC) at Fort Irwin, California; the Combat Maneuver Training Complex (CMTC) at Hohenfels, Cermany; the Joint Readiness Training Center (JRTC) at Fort Chaffee, Arkansas; and the Battle Command Training Program (BCTP) at Fort Leavenworth, Kansas. Their purpose is to provide Army units the most realistic combat training possible.

The NTC opened in 1981 and since then has afforded Army units the opportunity to train in a realistic combat environment to include a live fire exercise area. The CMTC, when completed, will provide a similar realistic, stressful training experience of a 4-day exercise for close combat heavy battalion task forces in Europe. The CMTC will not have a live fire capability. As a CTC, the JRTC will fill a void for training non-mechanized infantry battalions of the Active Army, Reserve Component, and National Guard. This training will prepare units for war under low- to mid-intensity combat conditions.

Directorate of Army Ranges and Targets (DART)/CTC, as the instrumentation acquisition manager for these CTC, develops acquisition plans and documents for instrumentation and manages the instrumentation procurement effort from concept to fielding.

INTRODUCTION

Short of actual combat, the CTC, provide the best training found anywhere in the world and the Army's leadership is fully committed to ensure their continued support and operations. In short--improved technology will permit us to better train as we will fight.

The CTC program is being developed to meet the Army's need for realistic battalion task force through brigade-level operational exercises that teach soldiers, leaders, commanders, and staffs the lessons of combat before the first battle of the next war. The Training and Doctrine Command (TRADOC) developed the CTC in keeping with its mission of "preparing the Army for war." The four CTCs for the Army are the National Training Center (NTC) at Fort Irwin, California; the Combat Maneuver Training Complex (CMTC), at Hohenfels, Germany; the Joint Readiness Training Center (JRTC), at Fort Chaffee, Arkansas; and the Battle Command Training Program (BCTP) at Fort Leavenworth, Kansas.

NATIONAL TRAININC CENTER

The NTC, which began operations in 1981, provides training for heavy maneuver battalion task forces stationed in the continental United States (CONUS). Brigade headquarters evaluations will begin in October 1987 ramping-up to the evaluation of a three battalion task force brigade by fiscal year 1990. A 14-day NTC training rotation provides each battalion task force 10 days of force-on-force exercises using multiple integrated laser engagement systems (MILES) and 4 days of live fire exercises against a regimental-sized opposing force (OPFOR) permanently stationed at Fort Irwin. The NTC presently trains 28 battalion task forces during 14 rotations annually. Each CONUS-based battalion (including National Cuard roundout battalions) trains at the NTC approximately every 18 months. The shift to brigade-level training will increase this tempo to 36 battalions during 12 rotations annually.

COMBAT MANEUVER TRAININC COMPLEX

The CMTC, scheduled to begin operations in 1991, will provide annual 4-day training rotations for 56 U.S. Army Europe (USAREUR) heavy maneuver battalions. A permanently stationed mechanized infantry battalion, with augmentation from the training battalion's parent unit, will provide a regiment (minus) OPFOR. Space limitations preclude live fire at the CMTC.

JOINT READINESS TRAININC CENTER

The JRTC will begin operations in October 1987 with full implementation scheduled for 1991. It will train Active and Reserve light forces (Light Infantry, Airborne, Air Assault, Ranger, Special Forces) for low- to mid-intensity conflict. Light battalion task forces will go through 10 day JRTC rotations annually; some rotations will be "exported" for units with terrain/climate-peculiar wartime missions such as Alaska. JRTC training will focus on force-on-force training, using MILES, against an OPFOR tailored for the training unit's specific mission.

BATTLE COMMAND TRAININC PROCRAM

The BCTP concept provides a training opportunity in a command post environment according to the wartime mission for division and corps commanders, their battle staffs, and major subordinate commands.

COMBAT TRAININC CENTERS

Combat Training Centers are distinguished from other Army training facilities by the presence of four unique capabilities:

(1) A permanently assigned, doctrinally proficient, impartial group of observer/controllers.

(2) A dedicated, realistic OPFOR.

(3) A superior training facility providing minimum restrictions on maneuver and maximum realism in simulation of realistic battlefield conditions.

(4) An unobtrusive system of instrumentation to collect data on training events and record, edit, and display this data for immediate training feedback and for longer-term analysis. The instrumentation system also assists in exercise command and control.

OBSERVER/CONTROLLERS

The observer/controllers for all CTCs will be provided by TRADOC, the "keeper" of the Army's warfighting doctrine. Comprised of experienced officers and NCOs, observer/controllers are assigned to each training unit from the battalion level down to platoon level. They are supplemented by training analysts who monitor the training via the instrumentation system. Together, these teams observe, control, and assist the training unit by interpreting battle events, determining unit performance strengths and weaknesses, and providing feedback through field after-action reviews (AAR) following each tactical mission and a rotation-comprehensive take home package. Observer/controllers also provide battlefield realism by playing the roles of the training unit's higher, supporting, and adjacent headquarters, and by simulating battlefield situations for which no automated system has been fielded.

OPPOSING FORCE

The OPFOR provided for each CTC will be sized, trained, and equipped to provide the training unit a realistic and challenging foe, accurately portraying the battlefield threat. Maximum accuracy in replicating Soviet and/or Warsaw Pact vehicles, weapons, and tactics is sought; engagement simulations of Soviet weapons systems have realistic range, accuracy, and probability of hit/kill. Elements of the OPFOR not in direct contact with training units will be replicated by an integrated system of simulators and computer-driven battle simulations.

SUPERIOR TRAINING FACILITY

The CTC will provide the best replication possible of the sounds, sights, and stresses of the battlefield. Within legal and environmental constraints (minor for NTC and JRTC, significant for CMTC), training units will be granted maximum freedom to maneuver, prepare positions, emplace and breach obstacles, and employ smoke. Besides MILES, for the direct fire battle, a comprehensive system of battlefield simulators will replicate the full spectrum of battle effects, including nuclear and chemical, indirect fire, air-delivered munitions, mines, and electronic warfare. Simulators will provide realistic physical cues and automatic casualty assessment, and will interface with real and/or training detection and monitoring systems.

INSTRUMENTATION

The purpose of CTC instrumentation systems is two-fold. The primary mission is to collect, edit, and display a complete and objective record of each training mission to provide the training unit feedback on its performance in near real time. The secondary mission is the provision of a historical and relational data base to support analysis of training, training development, and combat development. The instrumentation systems are built around position location/event reporting systems that allow training analysts to "see" the battle through computer graphics displays. The location of units and vehicles, weapons firings, casualties, obstacles, operations graphics, and the full spectrum of battle events are available to the analyst as these events occur, in a selectable range of detail. Live video of the battlefield and continuous monitoring of tactical communications give the analysts the fullest possible picture of the battle. Raw data is edited and packaged for immediate replay to the training units in AARs, which are also recorded to "capture" the observer/controllers' critiques. Upon completion of a rotation, units will be provided a complete data history of their training experience. In addition, computer work stations will be available at home stations for review of all or any part of the unit's rotation at the CTC.

BUILDING AND EXPANDING THE COMBAT TRAINING CENTERS OVER THE NEXT 10 YEARS

At the direction of the Army's senior leadership, an extensive analysis of the future direction of the CTCs was conducted during 1986. Since the CMTC and the JRTC are both still in the developmental stage, emphasis focused on improving and expanding the training capabilities of the NTC and applying these improvements, where appropriate, to the three new CTC. Key points of the approved NTC architecture were:

(1) Expand the NTC training rotation to a full three-battalion brigade.

(2) Expand the training analysis and feedback system to include not only combat units, but combat support, combat service support, and command and control elements.

(3) Improve data collection and analysis capabilities to increase the training benefits of the NTC experience.

(4) Accelerate development of realistic battlefield simulations.

(5) Increase the use of technology to reduce manpower requirements of improved CTC training.

NTC INDUSTRY DAY

Upon approval of the CTC concept by the Chief of Staff of the Army, the Army began developing specific requirements to achieve these goals. Realizing that there were no readily available technical solutions to many pieces of the puzzle, the Army hosted "NTC Industry Day" to identify the broad needs of the CTCs to industry, while emphasizing specific NTC requirements and soliciting industry's assistance in finding solutions to those needs. Hosted by the NTC at Fort Irwin in early May, Industry Day brought the training developers (TRADOC), the materiel developers (AMC), and the NTC users (FORSCOM) together with industry to define and discuss the Army's needs in an open give-and-take forum.

NTC REQUIREMENTS

Specific needs identified by the Army and discussed at NTC Industry Day included:

(1) A field digital data communications system for observer/controllers to improve both data collection and exercise control functions.

(2) The use of robotics/artificial intelligence (AI) and computer battle simulations to provide NTC training brigades a doctrinally-sized motorized rifle division (MRD), challenging OPFOR without the prohibitive manpower costs of a one-for-one replication. Computer simulations were also identified for replicating higher, adjacent, and supporting units of the training brigades.

(3) Improved battlefield realism from enhanced tactical engagement simulations including a shoot-through-obscuration capability, a realistic automated system of replicating indirect fires and other area weapons effects, and the accurate replication of the threat's electronic warfare capabilities, including radio, radar, jammers, and other emitters.

(4) Expansion of the instrumentation system's capabilities for training feedback and analysis through an advanced position location and event reporting system, and a fully interactive live fire training system featuring three-dimensional, maneuverable target arrays capable of "look back and shoot back" at training forces.

Although defined for NTC use, each of these capabilities is equally applicable, although on a smaller scale, to the JRTC and the CMTC. Each capability is discussed in detail in succeeding paragraphs.

FIELD DIGITAL DATA COMMUNICATIONS SYSTEM

Currently, much of the tactical skill of the observer/controllers is lost to the training analysis effort because there is no systematic, automated means to enter observer observations and evaluations into the NTC's computer data base. Time constraints and the field environment make written records inherently sketchy; even so, the bulk produced makes the analysis of even a few rotations' worth of information a tedious, unrewarding task. The provision of a digital data communications link between the observer/controllers in the field and the NTC Operations Center computer system would capture this irreplaceable but highly perishable information in a readily retrievable form. Using a portable data entry keyboard or similar device, the observer/controller could prepare preformatted reports quickly in the field, and then either send them immediately to the Operations Center or store them for submission during lulls in the battle. A free-format capability would allow brief narrative comments to be captured in the same way. By providing a two-way digital communications link, the system would allow the training analysts in the Operations Center and in the field to exchange information without relying on slow, inefficient voice radio. The field data communications equipment will be rugged enough for the harsh environment of all CTC's, light enough for man-portable use (especially for JRTC), and capable of operating on internal batteries or tactical vehicle power.

ROBOTICS/ARTIFICIAL INTELLIGENCE

The current OPFOR replicates a Soviet motorized rifle regiment (MRR(-)) on essentially a one-for-one basis, which in past NTC battles has proven to be a formidable opponent against lone U.S. battalion task forces. As the NTC expands to brigade level operations, replicating the MRD required for doctrinal realism on a one-for-one basis would require far more troops at Fort Irwin than the Army's force structure will allow. Therefore, the Army is seeking to replicate an MRD with technology capable of acting and reacting like men. OPFOR soldiers will still be required to be in direct contact with training units to provide the human interaction of close combat, but even here robotics or AI can be used to reduce crew requirements in fighting vehicles. For follow-on and rear area OPFOR elements, it is desirable to reduce the human element to the minimum, perhaps to only one vehicle in a unit. These robotic forces must be capable of realistically moving, detecting and reacting to threats, and providing self-defense with tactical engagement simulations. Visual, thermal, and electro-magnetic signatures will match those of the real forces being replicated.

COMPUTER BATTLE SIMULATIONS

Much of the training of commanders and staffs at the CTCs will center around their interaction with other U.S. forces: higher headquarters, adjacent combat units, and support elements. At the NTC, those forces not physically present are represented by members of the Operations Group who "role play" the appropriate interfaces. Information and support is provided to or withheld from the training unit by the role players in accordance with the exercise scenario. The deficiency of this system is that no scenario can provide for the unexpected. As exercises proceed, the scenario runs the risk of bearing less and less resemblance to the reality of combat on the ground. To bring consistency and coherence between the real and notional battles, computer-driven battle simulations, coordinated with the actual maneuver through the instrumentation system, will replace the current scenario structure as the source of notional events. Programmed prior to the exercise with real and notional U.S. and OPFOR dispositions and missions, simulations will draw from the real battle through the instrumentation system to generate a realistic notional battle involving the higher, flanking, and supporting units on both sides. Information derived from the simulation will be supplied to actual units as it would be in combat. Additionally, simulations will support command post exercises (CPX) of other units concurrent with the maneuver forces' training.

ENHANCED TACTICAL ENGAGEMENT SIMULATIONS

The direct fire battles at NTC are currently resolved using MILES, which uses eye-safe laser technology to simulate the effects of tank cannon, antitank weapons, and small arms with a credible degree of realism. Soldiers, vehicles, and aircraft can "kill" and be "killed" consistent with the capabilities of their weapons and their own tactical skills. Effective as the current MILES is, it does have significant flaws: it cannot shoot effectively through dust, smoke, fog, or other obscurants; it does not realistically portray the effects of range on antitank guided missiles' time-of-flight; and MILES cannot be easily adapted to different types of vehicles or to changes in weapons systems performance or vehicle vulnerability. Any future new system or enhancement to MILES must resolve these shortcomings by providing:

(1) The capability to effectively engage through battlefield obscuration out to the effective range of the weapon's limited visibility sighting system.

(2) An accurate replication of the variance in tracking times of antitank guided missile ranges.

(3) The capability to rapidly adjust engagement and detection criteria to vehicles and weapons systems of varying performance without modification to the systems.

INDIRECT FIRE/AREA WEAPONS EFFECTS

While the direct fire battle is effectively replicated by MILES (notwithstanding the shortcomings discussed above), the simulation of artillery, mortars, mines, nuclear and chemical weapons, and air-delivered munitions still relies almost exclusively on observer/controllers throwing pyrotechnics and assessing casualties manually, supported by nothing more sophisticated than a reference table of prescribed effects. Software to compute and display the effects of these weapons on the macro level is already available at the NTC; what is needed is a means of translating computer graphic displays into timely, realistic simulations of these effects on the ground. Simulators must provide realistic sight and sound cues to the training soldiers, with real time casualty assessment similar to MILES. The delivery of indirect fires must be as "transparent" as possible while providing the appropriate signatures to counterbattery target acquisition systems. Mine simulators must be capable of being detected and neutralized by standard Army equipment, or training equivalents. Nuclear and chemical simulators must interact with available detection and monitoring systems and require proper procedures for decontamination. All simulators will interface with the instrumentation system for control and casualty assessment purposes.

ELECTRONIC WARFARE (EW)

To fully prepare U.S. units to operate in the electromagnetic environment of the modern battlefield, units must be trained against the full array of Soviet electronic warfare assets, and to attack the Soviets' own command and control, communications, and intelligence systems. The present NTC OPFOR has only a limited capability to meet this requirement. Communications are based on U.S. equipment and procedures. Existing EW assets are limited to one or two jammers, direction finders, and radars. Fully exercising a U.S. brigade at the NTC requires that the OPFOR be capable of effectively representing the EW signature of an MRD, to include radio nets, radars, jammers, direction finders, and other EW emitters.

ADVANCED POSITION LOCATION SYSTEM

The key element in the instrumentation concept for the CTCs is the capability to capture the critical events of training exercises as they happen, and store them for later review and analysis. This is accomplished at the NTC by an existing position location and event reporting (PL/ER) system, which collects the location, weapons firings, and simulated hits and kills of up to 500 instrumented vehicles, aircraft, weapons, and individuals. These data are reported through a series of radio transponders and interrogator stations to the Operations Center computers where the information is processed for real time display and stored for later retrieval. Information updates are obtained from the instrumented systems at intervals ranging from 5 seconds for manpacks to a tenth of a second for high performance aircraft. The deficiencies of the current system are: it is limited to a maximum of 1023 instrumented players; it is dependent on line-of-sight from each instrumented system to interrogator stations, which effectively limits it to extremely open terrain; and its time constraints (359 updates per second) limit collection and transmission. Taken together, these shortcomings make the present system unsuitable for either the expanded NTC, or the two new centers. Requirements for the next PL/ER system for the CTCs include:

(I) The capability to track a minimum of 2000 instrumented players.

(2) The capability to operate in all conditions of weather, vegetation, and terrain, providing effective coverage of maneuver areas.

(3) Significantly increased data collection and transmission capability.

INTERACTIVE LIVE FIRE TRAINING SYSTEM

The NTC's live fire exercise is unique, not only among CTCs, but throughout the Army as a whole. It is the only place in the Army where the full range of a battalion task force's firepower, including artillery, attack helicopters, and close air support, can be employed without administrative safety restrictions. Defending against an array of 1018 silhouette targets representing an MRR in

the attack, the task force commander selects, prepares, and fights from his battle position as he would in combat, unhindered by safety fans or white-helmeted range officers. The observer/controllers with the task force interfere only when safety compels them to; maximum control is left to the unit's leaders. The result is a realistic, stressful, and challenging training experience unparalleled in the U.S. Army.

To increase the training benefit of the range, however, it is necessary to increase the fidelity of the targets' replication of the opposing force and to provide more effective feedback by obtaining a target's "eye view" of the training unit's performance. Envisioned is an array of three-dimensional targets, providing realistic firing from the air and the flanks. The array will be capable of real or simulated maneuver in response to the defender's actions as it advances. A down-range video system will give a view of the U.S. position from the MRR's perspective; flaws in deployment or fire control can, therefore, be captured for review. A tactical engagement simulation system will allow exposed defenders to be engaged and "killed" by the MRR, as is done in force-on-force today. In sum, the provision of a totally interactive target system capable of fire and maneuver will raise the tempo of the live fire exercises to that of the force-on-force maneuvers, while retaining the mental and emotional stress of live ordnance.

CONCLUSION

The National Training Center is the Army's premiere maneuver unit training system. Its contribution to the combat readiness of the Army is without doubt. To continue to expand that contribution as the NTC expands, and to export that knowledge to the JRTC and the CMTC as they become reality is the mission that the training development community faces. In an environment of resource scarcity, with skilled people the scarcest of all, the intelligent exploitation of technology becomes the key to accomplishing that mission. The Army has created a vision of the future for the CTCs and invites industry to join in mapping the road to that future. Getting there is the challenge and the opportunity that lies ahead.

ABOUT THE AUTHOR

Colonel Walter C. Zaremba is Director, Directorate for Army Ranges and Targets/Combat Training Centers, U.S. Army Training Support Center, Fort Eustis, Virginia. He holds BA and MA degrees in Political Science and a Juris Doctor Degree. He has 29 years of experience in the military. He is a graduate of the U.S. Army Command and General Staff College and the Army War College.

Since his assignment to DART/CTC, COL Zaremba has been responsible for the conceptual and technical design, systems integration, life cycle maintenance and training instrumentation support for the Combat Training Centers. He also directs and manages the Army Range Modernization Program.

CATIES

AN INNOVATIVE SOLUTION TO A TRAINING CHALLENGE

James B. Hollis Jr., LTC, Ret.
LB&M Associates, Inc.
111 C Ave, Suite 200
Lawton, Oklahoma 73501

Joseph R. Miller
Motorola, Inc.
Government Electronics Group
2100 East Elliot Road
Tempe, Arizona 85282

ABSTRACT

Problem: US ground combat forces currently have no way of rapidly and accurately simulating and assessing the effects of artillery and other indirect and area-effects weapons during training exercises.
Solution: The Combined Arms Integrated Evaluation System - (CATIES) simulates and helps measure the effects of conventional and tactical nuclear indirect fire support, nuclear - biological - chemical (NBC) contamination, and mine warfare. CATIES was developed to meet the Army's longstanding need for an indirect fire training device which would complement and interface with the direct fire Multiple Integrated Laser Engagement Simulation System (MILES). It also has the potential to simulate the lethal and suppressive effects of Navy and Marine sea and air delivered munitions and Air Force munitions delivered during close air support and air interdiction missions.
Application: CATIES applies to all combined arms, force-on-force training from small unit exercises to major joint training exercises worldwide. With CATIES, the total Army and Marine Corps forces - combat, combat support and combat service support units, will be able to train to doctrine in a more realistic indirect, as well as direct fire training environment.
Technical Approach: CATIES uses modern spread-spectrum radio frequency technology, employing pseudo-ranging, time-division multiplexing and surface acoustic wave signal processing techniques. The system can simulate up to 50 different effects per second which allows the replication of a multitude of indirect battlefield effects. Variable "hit" and "near miss" area sizes and shapes, in conjunction with expected fractional damages and casualties from approved munitions effects manuals, and unique audio-visual effects, ensure realistic battlefield training. Direct interface with MILES-type direct fire simulation systems provides an integrated solution to the indirect fire training problem. CATIES consists of three basic elements; 1) a Master Station, which receives voice or digital data from a fire direction or support element and transforms it into digital timing and weapon data. This data is transmitted to 2) Actuators which in turn retransmit this data at precise time intervals to 3) Appliques located on vehicles, personnel and/or terrain features. The Player Detection Devices respond to the arrival time of the transmitted pulses, the weapon-munitions type, and the target type and size. The capability to relay data through other Actuators and electronic line-of-sight technology assure wide area coverage, with optimal message routing determined by the Master Station.

INTRODUCTION

As the approaching dawn peaks across the desert landscape of the National Training Center a US Army mechanized infantry task force commander searches for the tell-tale signature of the attacking enemy force. Although he is confident in the ability of his TOW gunners, his tankers, and his other direct fire systems to acquire and successfully engage the tanks and other armored fighting vehicles of the atttacking Soviet regiment, certain nagging doubts continue to haunt him.

In the calm before the storm he remembers his introduction to combat as a young company commander in a far off corner of the world - a night when his rifle company experienced the mortar and rocket prelude to a North Vietnamese Army ground attack. He remembers the deafening explosions, the beehive sounds of whinning shrapnel, the pungent smell of exploding munitions, and the call of his wounded for help. He recalls the almost paralyzing effects on his ability to remember what he should do next, his inability to talk to his radio-telephone operator over the roar of battle, and his near total loss of control during the first few moments of the actual ground attack. He remembers with disgust his inability to describe his own situation to an inquiring battalion commander because only one of his three platoon leaders was on the radio. Finally, he recalls how long it took for the men in his platoons to resume good firing positions and to deliver well aimed fire at the fleeting targets presented by the attacking forces. The task force commander's concern increases.

Suddenly the commander is shaken from his early morning thoughts of his first taste of combat by calls from his scout platoon to his operations element. The Bradley Infantry Fighting Vehicle (IFV)-equipped scouts positioned forward and on the flanks of the defending company/teams are reporting the initiation of the regimental attack. Dust clouds rise in the distance as the enemy tanks and BMP's approach the carefully planned and prepared obstacles of the defending task force. Enemy reconnaissance elements are already beginning to

probe his outer defenses and are attempting to determine where and how to penetrate his battle positions.

Knowing Soviet doctrine as he does, the task force commander expects to receive a devastating volume of preparatory fires by enemy divisional and regimental artillery groups. (According to a recently published Field Artillery School "white paper" on Warsaw Pact artillery, a task force in a similar situation could receive as much as 20 to 50 minutes of preparatory fires. The preparation fires could include up to 23,000 artillery, mortar, and rocket rounds, or over 2300 tons of HE, fragmentation, illumination, and smoke rounds). On this day at the NTC nothing occurs except a couple of fire marking teams strolling through his company/team positions throwing artillery simulators and subjectively causing casualties with the "God Gun", a master laser gun used to turn on the "hit" mechanism in the Multiple Integrated Laser Engagement System (MILES) worn by solders and strapped on selected combat weapon systems.

The nearby task force fire support officer announces that he is beginning to monitor calls for artillery fires on his TACFIRE Variable Format Message Entry Device (VFMED) - calls for fire which are being transmitted digitally from the infantry and armor fire support team (FIST) chiefs and forward observers (FO's) to the direct support artillery battalion's Fire Direction Center (FDC). He waits expectantly for the impact of the supporting artillery. Once again, very little indirect fire simulation occurs. In a very unconcerned manner the opposing forces (OPFOR) quickly breach the mine field and wire barriers which protect his positions. The calls for "final protective fires" quickly follow and are met with the same response - a complete absence of battlefield effects.

For the next four hours the battle swirls about him as he maneuvers his defending forces and as his direct fire elements give a good account of themselves. Because he has trained his forces well for the direct fire battle, and because he has successfully anticipated the flow of the maneuver battle and insisted on the preparation of alternate and supplementary fighting positions, his task force carries the day.

The after action review addresses in glowing terms the ability of the task force commander and his staff to understand the brigade's scheme of maneuver and plan of fire support, prepare and distribute combat plans and orders including a vivid description of the task force commander's intent, anticipate the maneuver of the enemy forces and to exercise initiative within the scope of the brigade commander's intent, react to unexpected threats and opportunities, and to engage enemy forces with their direct fire weapons.

Unfortunately, the after action review contains very little objective and factual information about the effects of friendly and enemy artillery, mortars, mines, and chemical weapons delivered by either side during the battle. None of the NTC controllers really know what effects the enemy's regimental and divisional artillery groups' prep might have had on the defending friendly forces. No one really knows whether the task force's supporting artillery could have suppressed enemy gunners and forced enemy tank commanders to "button up" thereby degrading their target acquistion and engagement capabilities. No one really knows if we could have delayed and suppressed follow-on and uncommitted forces and prevented the enemy from "piling on" in the vicinity of the FEBA. In effect, no one really knows if certain fundamental aspects of our AirLand Battle doctrine are valid at the brigade and battalion levels of conflict.

Throughout the task force commander's 14 days of training at the training center the story is the same - less than effective means of crediting the maneuver, fire support and engineer communities for effective indirect fire and barrier planning and coordination, and practically no experience for his troops in preparing physically and psychologically against enemy weapons systems (artillery, mortars, and rockets) which outnumber our own by ratios of 5 or 6 to 1, or more . And so the questions remain. How prepared are the task force commander's troops for war, **really**? How competent are his forward observers and fire support officers? How good are his engineers at emplacing mines and preparing obstacles? How ready are his troops to engage in offensive and defensive chemical operations? Can we delay follow-on forces and suppress enemy artillery and air defense firing elements? **How sound is our doctrine?**

THE TRAINING REQUIREMENT

Currently the US Army has no way to realistically simulate and to accurately measure the effects of area weapons such as artillery, mortars, mines, chemical and certain aerially delivered munitions. Specifically:

- The disruptive artillery fires are frequently notional and, at best, simulated by manpower intensive and less than timely fire marker teams tossing unrealistic artillery simulators that seldom represent the coverage and never the suppressive effects of indirect fire munitions.

- Chemical attacks are rarely a surprise in training exercises, and because they are usually notional (the NTC does use CS or tear gas), there are no objective methods to sense and penalize failure to meet accepted chemical defense postures.

- Employment of barriers in most training situations is often notional and does not delay or canalize the oppposing force realistically, again because there are no objective methods to sense and penalize failure to meet accepted procedures. Because of a lack of realistic simulation of the lethal and audio-visual effects of indirect fire enhanced barriers, opposing force elements are not suppressed and slowed as they might be in combat.

- Aside from the objective assessment made possible by fielded and emerging direct fire training engagement simulation systems, personnel and equipment casualties are determined by subjective, inconsistent estimates, usually well after-the-fact.

A recently published study by Rand Corporation's Arroyo Center analysis group describes the indirect fire simulation and assessment system presently in effect at the National Training Center.

> "During force-on-force battle simulations at the NTC, artillery fires are represented on the Core Instrumentation Subsystem. Unlike direct fire, however, the inputs to and outputs from the computer must be accommodated manually, and battle damage assessment relies in part on subjective judgements.
>
> Calls for fire pass up the normal fire direction system from the forward observers (or whoever is calling for the mission) to the artillery operations center. (Most training units use TACFIRE systems, and a few still use voice radio.) There the mission will be "fired" by order to the firing battery. Some requested missions are not fired, owing to priority allocation of fire. The fire order is also passed to the artillery analysis team in the Central Instrumentation Facility, where the firing data are entered into the computer (tube location, target location, rounds fired, etc.). At the same time, fire markers or observer/controllers are directed by radio to mark the fires using pyrotechnic simulators at the target location.
>
> The computer displays the mission, but the analysts in the facility and the field observers or fire markers manually carry out the damage assessment. An impact box of standard form is shown on the display. If the analyst watching the unit sees the box cover a part of the unit, or if the O/C or fire marker in the field, directed to the location of the "impact," finds forces near it, they can agree, by radio link, to the proper battle damage assessment.
>
> Standard tables are used to determine the damage to be assessed by a given mission (e.g. 24 rounds of high explosives) against a given target (e.g. a dismounted platoon in prone positions). The assessed artillery results are not made a part of the computer record, although the observer/controllers may make a field note of the results. The artillery analysis team records each fire mission in a log. That log shows the time of fire, the caller (if known), the type of mission, the target location, and whether the mission was good (hit an enemy target), no good, or has hit friendly forces. The log does not contain information about the target or the battle damage assessment. These manual logs are retained in the artillery section for a few months and then discarded. A similar system exists for OPFOR artillery play."

The ability of commanders at all levels to achieve maximum, synchronized combat power at the proper time and place on the battlefield is dependent upon the extent to which they are able to train themselves and their subordinates in peacetime. As alluded to above, such training is currently hampered by a training environment which neither portrays the contribution of fire support to the combined arms effort, nor represents the effects of friendly and enemy fire support on combat operations at all tactical levels.

With the arrival of MILES and more recently the Army's MILES-compatible Air Ground Engagement Simulation-Air Defense (AGES-AD), the maneuver, air defense and air support components gained a more realistic and effective training system to simulate the effects of direct fire. The line-of-sight characteristic of these systems makes them ineffective in the simulation of indirect fire munitions, thus, the indirect fire support elements can not realistically participate. As a result, training of the maneuver elements, who benefit most from an understanding and appreciation of the effects of both friendly and enemy fire support is less than adequate.

The absence of a means to simulate objectively the effects of indirect fires has produced at least three distinct training deficiencies:

- Maneuver unit commanders often under-emphasize the use of indirect fires because of the unrealistic, subjective and time consuming nature of current simulation systems. This leads to a lack of appreciation for the contribution of artillery and mortars on the battlefield. For example, in a letter in the March-April 1986 INFANTRY magazine an armored cavalry squadron commander stated:

  > "...We have been on more than a dozen REFORGERs over the past ten years and I can tell you that artillery is virtually worthless to the tactical commander in these exercises. This is because the cumbersome system used to allocate credit for artillery is unworkable. Many commanders stop using artillery because they will never get credit for it, and there are other things they can do with their time..."

- Combat arms, combat support and combat service support elements train in an environment devoid of the suppressive effects of the enemy fire they are most likely to experience in combat, i.e., air and surface-delivered indirect fires.

- The individual soldier, even in the maneuver battalion task force, cannot experience in training the surprise, destruction, disruptive and suppressive effects of indirect fires.

To train effectively, the total force needs to be able to train in a more realistic indirect fire, as well as the more realistic direct fire environment made possible by the MILES-type training simulators. To quote from *The Posture of the United States Army for Fiscal Year 1987*;

"...While MILES has provided unparalleled opportunities for realistic, two-sided, tactical training world-wide, true combined arms tactical engagement training is being sought. Efforts to incorporate the simulation of artillery and mortar indirect fire, mines, and NBC area weapons effects into MILES exercises will improve tactical engagement training."

Another document, the U.S. Army approved *Fire Support Mission Area Analysis* (MAA) states the need for realistic, effective and safe indirect fire simulation and evaluation in training exercises.

"... a large training gap exists in the need for devices and methods to realistically play indirect fire systems in the MILES exercises both at the National Training Center and other installations having MILES equipment."

The MAA further specifies the need for ----

- A flash, bang, smoke cue that gives training participants an appreciation of the lethal and suppressive effects of indirect fires and causes them to take proper preventive measures to survive and carry out the mission.

- An automatic casualty assessment system which alleviates the need for fire markers and assesses casualties based on the type and coverage of munitions employed and nature of the targets in the affected area.

The Solution is a system which simulates the contribution of Army, Navy, Air Force and Marine fire support to the AirLand battle, portraying the effects of indirect fire support. A training system which integrates and simulates these effects should --

- Capitalize on and complement existing and developmental MILES-type direct fire engagement systems.

- Provide realistic battlefield effects.

- Provide realistic training for the total combined arms force.

There have been several attempts over the past ten years to get beyond the old fire marker and subjective assessment operation, but technology and safety restrictions have limited the development of cost-effective solutions. However, recent advancements in micro-chip and radio frequency technology, particularly the miniaturization, increased capacity, and reduced cost of key electronic components, permit applications of unique combinations of these components to meet this simulation need.

The Solution

CATIES meets the urgent training requirement for a complete fire support simulation and assessment system. CATIES will provide the capability to simulate the effects of conventional and tactical nuclear fire support, NBC contamination and mine warfare in combined arms, force-on-force training, from small unit exercises to major joint training exercises, worldwide. With CATIES, Army and Marine combat, combat support and combat service support forces will be able to train to doctrine in a more realistic indirect fire training environment that will include simulation of the lethal and suppressive effects of Naval gunfire, and Air Force, Navy and Marine Corps aerially delivered munitions.

CATIES: THE SYSTEM

Currently, MILES provides a means to judge the effectiveness of direct fire weapons on an opposing force. When MILES sensors on opposing force soldiers and equipment are activated by laser energy, they indicate either a "near miss" or "hit". A hit can be further categorized as resulting in either damage or destruction (wound or kill for personnel). The system takes into account the type weapon, tracking requirements and the nature of the target. In a parallel manner, CATIES employs radio frequency (RF) energy to activate a target sensor (Appliques) while taking into account weapon and munition characteristics and the nature and disposition of the target. The RF signal is not easily attenuated by dust, smoke or foliage; thus, it is better suited to simulate the effects of indirect fires, NBC and mine warfare. As depicted in Figure 1, CATIES has three primary components:

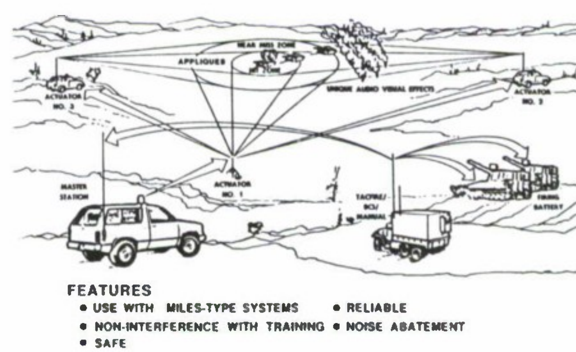


Figure 1 - CATIES SYSTEM

- The Master Station, which initiates and controls the system through the transmission of attacking weapons and timing data to selected Actuators.

- The Actuators, which transmit directly or act as relays for the transmission of weapons and timing data that cause the activation of appropriate Player Detection Devices.

- The Appliques, which sense Actuator transmissions of coded energy and provide indications of the effects of the simulated munitions on the targets.

Master Station

The Master Station, shown in Figure 2, consists of a micro-computer, receiver/transmitter, graphics display and necessary communications equipment to link with firing unit's fire direction facilities and fire support elements.

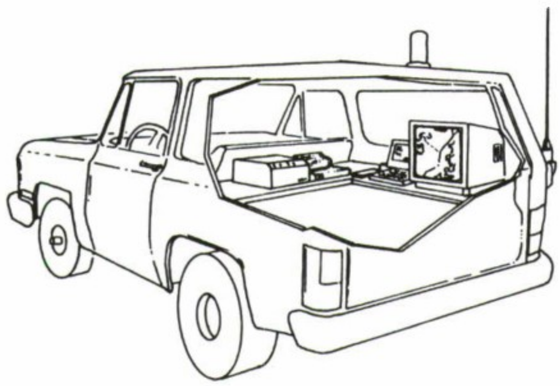


Figure 2 - Master Station (MCS)

Based on the target location, method of fire, and time, the Master Station computes the data required to cause each of at least three coded, omnidirectional, RF energy pulses to be transmitted through selected Actuators to intersect over the target location at precise time intervals. Considering electronic line-of-sight technology and using Actuators as relays, the system's range can be extended to over 100 kilometers.

Actuator

The solar battery powered remote Actuator (Figure 3 below), consists of a microprocessor-controlled receiver-transmitter, antenna, cabling, and an auxiliary communications device, all contained in an easily carried combination case.

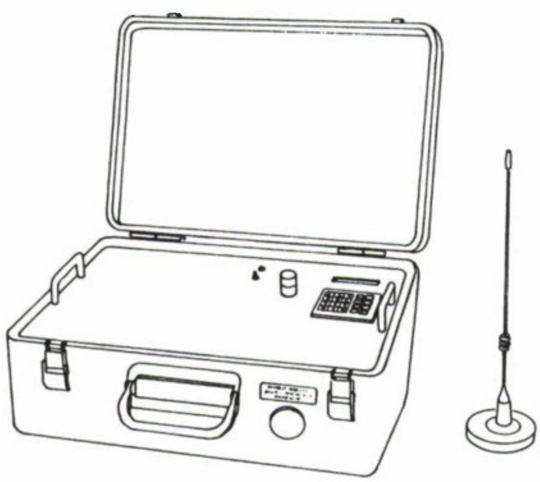


Figure 3 - Actuator

The Actuator receives the timing and weapons data from the Master Station , and transmits the coded radio frequency signal to Appliques positioned on personnel, equipment, and terrain features. The Actuator includes a keyboard and digital display used to input surveyed location data at time of emplacement, and to perform other routine functions such as self-test. At least three Actuators, each with electronic line of sight to the designated target, are required to activate an Applique. The maximum Actuator-to-target range is over 15 kilometers, and as stated earlier each Actuator is capable of relaying Master Station data to other Actuators in order to extend the operating range of the system and circumvent RF line-of-sight problems. The Actuator, once emplaced is designed for autonomous operation. Typically the Actuators will be located in vehicles, or when used in permanent training areas such as the National Training Center, on man-made structures such as small towers or platforms.

Applique

The Applique, depicted in Figure 4, is a receiver-decoder slightly larger than a cigarette package, and is placed on an individual soldier, vehicle or other appropriate object and linked to a flash-bang-smoke cue and MILES-type device. Receiving the appropriate, coded signals an Applique activates to indicate either a "hit" or a "near miss".

Figure 4 - Applique

Safe, nondud producing, indirect fire unique audio-visual effects will represent either result. A "hit" will actuate the MILES device for casualty or damage assessment. The Applique is programmed with a changeable "target plug" to represent a specific type of target and uses established effects probability data to determine the effects on that type of target.

Audio-Visual Cueing Device

A fourth component, not listed earlier as part of CATIES, but just as important, is a flash-bang-smoke producing device. It will complement the lethal effects simulation with appropriate audiovisual cues so critical to the soldiers affected by simulated indirect fires, chemical contamination or mine warfare.

OPERATIONAL CATIES

The Training Environment

CATIES is adaptable to large Advanced Collective Training Facilities (ACTF's) such as the Army's National Training Center at Fort Irwin, CA. or

to large training exercises which use civilian owned land and facilities such as the Army's annual Redeployment of Forces to Germany (REFORGER) exercise. It is also appropriate for use in highly confined and restricted Local Training Areas (LTA's) such as those found on or near posts in Germany and in the continental United States.

Set Up

Preparation time and effort varies in relation to the permanency of the training area, but in general "prepare-to-train" operations proceed as follows. Once the limits of the training area are defined, the Actuators are positioned where they provide coverage of the area of operations. Actuator locations must be precisely determined and recorded. The number of Actuators required is a function of the size and terrain characteristics of the exercise area. When an exercise moves over large expanses of terrain, such as during REFORGER, the Actuators can be moved quickly; however, to ensure continuous, electronic line-of-sight coverage for a brigade-size element, at least three, preferably five Actuators, must be in position and operational all of the time. Actuators can be operated by vehicle power or by an internal solar charged battery.

The Master Station is positioned where communications can be established with appropriate fire direction centers (FDC's), fire support elements, and the Actuators. The size of the organizations exercising and the size of their area of operations may require more than one Master Station Normally, one Master Station will be in communication with unit mortar platoon FDC's as well as supporting field artillery FDC's. When the Master Station relocates the micro-computer must be initialized by entering the type and location of all its associated indirect fire units, and the surveyed locations of its Actuators. Two Master Station operators are considered sufficient manning for continuous operation of an Master Station over a three-day exercise.

CATIES Appliques are placed on all appropriate personnel and equipment participating in the training exercise. The Appliques are initialized by inserting a target plug which identifies the Applique as a certain type of target (e.g. individual soldier, tank, infantry fighting vehicle, truck, etc.). The CATIES Appliques interface with the MILES sensor equipment worn by a soldier or affixed to a vehicle, allowing the audio and visual alerts within the MILES device to signal an indirect fire "hit". Additional distinctive tones and visual signals will be used to distinguish between direct fire and indirect fire "hits" or "near misses", and to cause the individual trainees to take appropriate action. The Applique can be deactivated and reset by the same controller who resets the MILES devices.

Operational Sequence

Following a single manually processed fire mission demonstrates how CATIES will be used in a tactical engagement simulation. Although this paper illustrates a manual solution, the fully developed CATIES will be capable of receiving and processing digital information.

- Indirect fires are planned and requested in accordance with current doctrine. The sequence to be used here starts with a fire request from an FO supporting a maneuver unit. The FO requests fires either digitallly or by voice means, over the appropriate field artillery fire net. As a minimum, the FO indicates the target location, nature of target and method of control. For this scenario the target is a platoon of tanks and an accompanying platoon of dismounted Infantry.

- The supporting field artillery 155mm howitzer battalion designates an available battery to fire the mission with two battery volleys of dual-purpose improved conventional munition (DPICM). The battery performs the required technical fire control operations at its FDC. The exercise controller in the FDC or one of the unit's FDC personnel sends the following information to the CATIES Master Station as soon as it is available.

  - Location of target.
  - Time of flight.
  - Shell-fuze type.
  - Number of volleys and number of tubes firing.
  - Radius of target.

- The operator at the Master Station enters this FDC data into the micro-computer which selects at least three Actuators that are in range and have electronic line-of-sight with the target. Then the operator awaits receipt of the time of "shot" from the FDC. When the time of "shot" is received, the Master Station operator enters it into his micro-computer.

- Based on time of flight, the computer calculates time of impact of the shot and time codes needed to transmit the RF signal through the Actuators to the impact area. At the calculated time the Master Station transmits the coded signal containing the type weapon and munition data to the selected Actuators (Figure 5). The Actuators process and retransmit the coded pulses to arrive at the target area in the proper sequence at the time of impact of the simulated indirect fire. The pulses are separated by very small, precise time increments which cause the proper effect on target Appliques. The Appliques decode the pulse timing to indicate either noneffect, actuation of Appliques to indicate a "near miss", or actuation of the Appliques to indicate a "hit" in accordance with JMEM-based probabilities. The time of each of the pulses is critical because the intersection of these pulsed signals at their time increments described above define the target area. In this example, the target area will be roughly elliptical, approximately 300 meters by 200 meters.

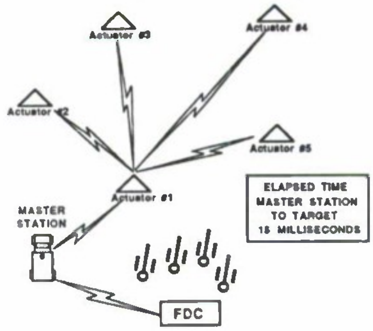
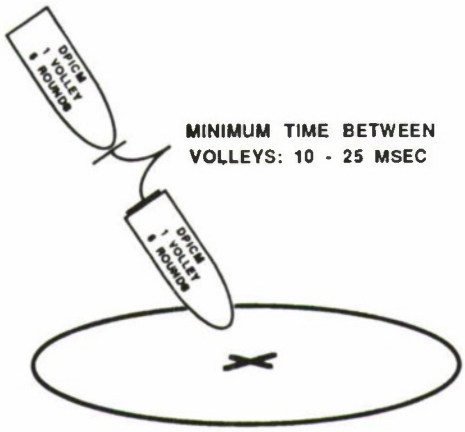


Figure 5 - CATIES Pulse Sequence

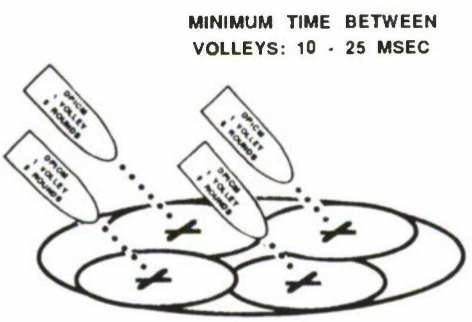
• The same procedure is followed for subsequent (in this case the 2nd) volleys (Figure 6). Multiple volleys on separate aim points can also be simulated (Figure 7).



MULTIPLE VOLLEYS ON SAME AIM POINT

Figure 6 - CATIES Simulation Capability

Flight times for indirect fire munitions are on the order of magnitude of tens of seconds with minimum time intervals between volleys from the same weapons of approximately 10 to 15 seconds. Thus, the minimum times indicated in each of the figures shows the responsiveness of CATIES to be more than sufficient to allow real time simulation of indirect fires. CATIES represents both flight times for projectiles and timing between volleys in real time to coincide with the simulated firing and impact of subsequent rounds or volleys.



MULTIPLE VOLLEYS ON SEPARATE AIM POINT

Figure 7 - CATIES Simulation Capability

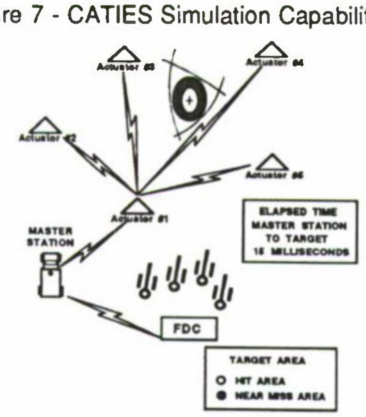


Figure 8 - CATIES Pulse Sequence

• As stated earlier, the target area is defined by the intersection of at least three RTD signals (Figure 8). Because these signals are omnidirectional, they naturally intersect at numerous points in the training area.The precise increments of time that each Applique will accept and process properly coded signals determines which of the intersections define the "near miss" area and which define the "hit" area (Figure 9).

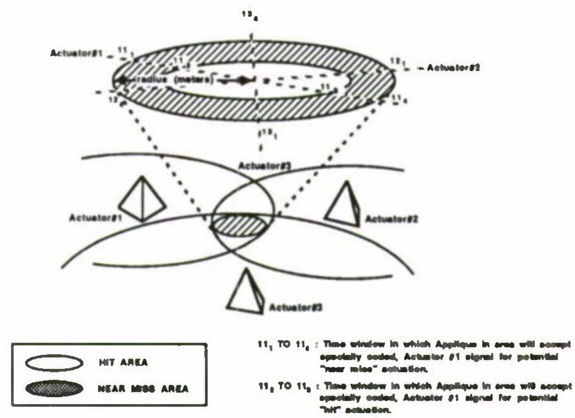


Figure 9 - CATIES Target Area Simulation

• An Applique in the "near miss" area will activate if it receives at least three separate properly coded signals within a specified period of time. When activated, the Applique will cause audio and visual cues to be emitted from devices that will indicate indirect fire-unique sounds and visual effects. An Applique in the hit zone is designated as a "hit" or "near miss" based on JMEM probabilities for the type target and munition fired. If designated as a "hit", the Applique will cause emission of indirect fire unique sounds and visual effects and cause the MILES device to emit "hit" audio and visual alerts. If no hit has occurred, the "near miss" alerts are emitted.

• When "shot" for the second volley occurs, the above procedure is repeated. If all or a portion of the tank/infantry target, having been alerted by "hits" and "near misses" from the first volley, is able to move out of the target area, then fewer "casualties" will result from the second volley.

• Exercise controllers reset MILES devices which have registered "hits" using the same procedures they use for direct fire activations. Personnel casualties can be assessed by using the same cards as are used for direct fire assessment.

SUMMARY OF CATIES FEATURES

CATIES possesses the following characteristics

• Indirect Fire Effects Assessment - CATIES provides a means of assessing the effects of indirect fires on the battlefield. The CATIES system uses JMEM-related probabilities for target damage based upon the nature of the target and the types and numbers of munitions employed.

• Timely and Realistic Indirect Fire Simulation - The effects of indirect fires in battlefield simulations with CATIES are simulated in real time through the use of radio frequency signals which accurately define the target area. No longer must trainers wait on fire marker teams to arrive at a target and attempt to define the boundaries of the target area with simulators. CATIES gives soldiers greater battle realism and awareness of indirect fires in their vicinity through MILES audio and visual cues and indirect fire-unique sound and flash devices.

• Savings in Manpower - CATIES interfaces directly with MILES, requiring no additional controllers for indirect fire engagement simulations. In fact, the elimination of need for dedicated indirect fire markers offers a significant opportunity for overall reduction in controller requirements. CATIES itself requires very few personnel and minimal training. Actuators can operate unattended, requiring only one or two personnel to move them and set them up. The Master Station can be operated by as few as two people.

• Offers Opportunity to Train to Doctrine, Worldwide - CATIES can be integrated into training exercises at all levels - from platoon through corps anywhere that MILES-type devices are used. For a platoon-size exercise, for example, the battalion mortar platoon FDC provides sufficient capability to answer forward observer calls for fire, thereby exercising the fire support system at the lowest echelon.

• Minimal Interference with Training - CATIES has no adverse impact on the training area and its operating considerations are virtually transparent to exercise participants. No vehicle cluster need be established between opposing forces, nor are other elements of artificiality needed with CATIES. Weapons effects simulators which create potential safety hazards are no longer required.

• Application to NBC and Mine Warfare - CATIES offers a capability beyond the fire support arena. The CATIES concept can be adapted for use in both the simulation of minefields, chemical and nuclear battlefield operations. The capability of the Master Station and its set of Actuators to cover (within range) up to 50 different areas per second gives CATIES the potential for continuous pulsing of areas to simulate the family of scatterable mines (FASCAM), conventional mine fields, chemical contamination (both persistent and non-persistent), and the downwind movement of contaminants. CATIES offers tactical engagement simulation for virtually the entire combined arms team.

ABOUT THE AUTHORS

Mr. Hollis is a recently retired Armor officer currently serving as a systems analyst with LB&M Associates in Lawton, Oklahoma. He is responsible for the operational and functional analysis and development of CATIES. His career in the Army included assignments as doctrine writer and maneuver tactics instructor at the Field Artillery School and the Command and General Staff College. He was an armored cavalry troop commander in Vietnam and Germany, and a tank company commander and battalion S3 in CONUS. He is a graduate of the U.S. Army Command and General Staff College, the U.S. Army War College Defense Strategies Course, and holds graduate degrees in Public Administration and Human Resource Development from the University of Oklahoma and George Washington University, respectively.

Mr. Miller is the Motorola staff engineer and project leader responsible for the design and development of a proof-of-principle system which simulates area weapons effects. Previous Project Leader experience includes the engineering development of the Portable Control System (PCS). System capabilities include command and control of Remotely Piloted Vehicles. He is also currently a Design Engineer for the hardware portion of AFDC (Automatic Formation Drone Control) including airborne range-to-range processing and ground station modification for multiplexed command and telemetry. Mr. Miller received Honors at Entrance and was on the Dean's List at Arizona State University. He also holds Tau Beta Pi, Eta Kappa Nu. In 1976, Mr. Miller was awarded the "Engineering Creative Design Display" at the University of Tennessee at Knoxville, by ASEE.

THE NEW TECHNOLOGY OF LARGE SCALE SIMULATOR NETWORKING: IMPLICATIONS FOR MASTERING THE ART OF WARFIGHTING

Lt Col Jack A. Thorpe (Ph.D.), USAF
Defense Advanced Research Projects Agency
1400 Wilson Blvd.
Arlington, Virginia 22209

ABSTRACT

Advances in several core technologies, particularly local and long haul networking, open up a new area in simulation: Large scale simulator networking (SIMNET). This has important implications for training warfighting skills as well as providing tools for other areas. These are discussed along with a description of new capabilities and future directions.

INTRODUCTION

It appears that the ability to plug together large networks of simulators is well within our grasp. Local area networking technology is established and can be purchased off the shelf for connecting perhaps hundreds of simulators at a given site. Long haul networking technology is maturing rapidly and will provide force-on-force gaming between sites. Microprocessors, the interchangeable muscle on network skeletons, grow in strength and drop in cost with each new generation every couple of years. And a fresh look at simulator design is making it easier to match the physical and performance characteristics of simulators to the needs of the combat team member.

These breakthroughs have far reaching implications for the field of simulation. For the first time we have the opportunity to attack the premier training problem of the military: How to master the art of warfighting.

WARFIGHTING

Modern warfighting is the most complex activity performed by man. It is rooted in each individual's performance with his single weapon system, support system, logistic system, administrative system, or whatever system he or she must operate as part of the broad machine of combat.

But its scope is far greater. It includes the coordination of that individual's activities with others in the crew, and that crew's interaction with other crews, their interactions with other larger teams of similar combat systems, and the team's interaction with combat support and services support of their own branch and their own service. It includes the interactions of combatants between branches (armor interacting with attack helicopters, for example) and interactions with other services (close air support). On the highest level, it includes cooperating forces of different nations and different languages interacting with each other on a common battlefield.

To be successful at warfighting combatants must master these interactions at all levels. As the implements of war change, the common denominator remains that people have to interact. This is the constant in battle. Training of this is training for teamwork, coordination, execution, orchestration of the battlefield. It is the essence of successful warfighting.

Up to now the United States has relied on field exercises to bring together the component skills needed for warfighting. In sports, these would be called the scrimmages or preseason games which exercise the whole team: the coaching staff, the equipment and conditioning staff, the spotters, the scouts, the front office, as well as the players on the field and on the bench. The need to exercise the whole team distinguishes this from other types of training: Training for team execution requires involvement and practice of the entire team under conditions representative of the contest.

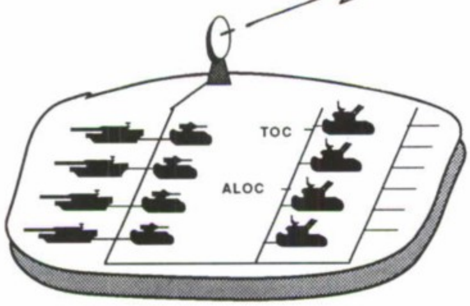
Exercises like thc the Army's National Training Center and the Air Force's Red Flag are examples of scrimmages practiced in the military. They are particularly good at creating the chaos that accompanies all large human enterprises, chaos which Clausewitz chose to characterize as the fog of war, the principal determinant of failure.

Yet even as good as these field exercises are, training with real combat equipment on ranges has limitations: Combatants are limited in how far they can push their systems because of safety, participation is limited in duration and frequency because of cost, and hardware is maintained at far better levels than what can be expected a short time into actual war.

Nonetheless these exercises are valuable. Units learn how to work together under stress, and leaders learn about the dynamics of team operations in chaos.

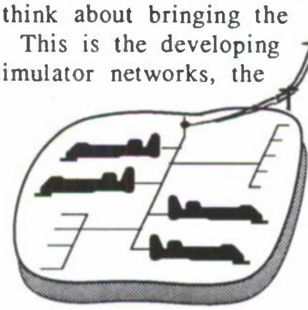
Further, the resident opponent or aggressor teams at these centers give us insight into the overwhelming importance of practice in the mastery of warfighting: The aggressors have become consummate, cunning warfighters as a result of the thousands of hours of practice they receive during their tour of duty as the threat force. They are formidable opponents. They have mastered warfighting.

This reinforces what we already know about how teams achieve mastery of their art, be it a sports team, an orchestra, an operating room team or a combat team: Massive amounts of practice is demanded. There is no substitute.



Local Area Network (LAN)
(2 - 200 Simulators)

If the bad news is that to build proficient warfighting teams we have to provide this kind of practice in large amounts with only a small proportion available from the field, then the good news is that recent developments in technology enable us to think about bringing the field into simulation. This is the developing area of large scale simulator networks, the



initial work being done in DARPA's SIMNET program (for <u>sim</u>ulator <u>net</u>working) in partnership with the Army and now the Air Force.

WHAT IS IT?

Large scale simulator networking encompasses the local and long haul nets which connect not only combat simulators but also all their command and control, logistics, administration, and other combat support and services support activities. It is a vertical as well as horizontal slice of the battlefield. Because it practices the entire warfighting team in simulation, all those who would fight in a real battle come to netted simulators and combat stations to fight. Both sides.

Combat typically is on real world terrain. Simulators have identical copies of terrain data bases resident in memory and exchange order of battle information via networking. The R&D version of SIMNET now fights on 50km x 50km battlefields, but this will be expanded to battlefields several hundred kilometers on a side in the near future.

The scale of battle is many times greater than what current simulations now enjoy. If R&D progresses at its current rate, networks will be

capable of connecting several hundred combat simulators, command, staff, and support elements in the near future. Several thousand personnel will be involved.

As an example, the SIMNET R&D project is developing test sites of joint airland battle forces. At one such site a tank heavy battalion sized task force will be supported by an aviation company of three scout and five attack helicopters, two elements of air to ground fighters, air defense, fire support vehicles, a scout platoon, a tactical operations center, tactical air control center, forward air controllers, admin/log center (with fuel, ammo, and maintenance vehicles), and artillery and mortars. This totals 44 M1 tanks, 20 M2/3 fighting

vehicles, 4 air defense vehicles, 4 fire support vehicles, 8 helicopters, 4 air to ground fighters, and miscellaneous M577 command vehicles, fuel, ammo, and maintenance trucks. These will be fully crewed combat simulators and elements.

This site will be netted to other sites for force-on-force combat. Friendly air support could come from one site, opposing artillery from a second, and reinforcing armor from a third, all fully interactive in real time.

But because of the very nature of networks and those simulators designed for them, the overall network does not have to be fought in one large conglomerate. Networks can be reconfigured into smaller non-interfering clusters of combat fought on different terrain patches under different conditions, all at the same time. As an example, a network of 100 simulators could be fought in one battle (e.g., 50 offense vs. 50 defense, 60 vs. 40, 10 vs. 90, or whatever is called for by the commander organizing the operation) or it cuold be broken down into two exercises (e.g., Battle #1 with 30 vs. 20 and Battle #2 with 25 vs. 25) or any other combination down to the lowest element of 100 separate, non-interfering single vehicle exercises.

These reconfigurations are managed with a microprocessor and take just a few minutes to arrange. Just as combat elements are task organized for a specific mission against a specific threat in real combat, exercises on networks are configured in conference call fashion to meet a specific need.

This dial-a-war way to task organize a network uses the same military chain of command as in combat. Warfighting operations here are the same. Operations orders are issued, forces are assembled, map reconnaissance is conducted, radio frequencies assigned, stores positioned on the terrain, preplanned artillery and air strikes coordinated, and so on. Crews mount their simulators and carry out their missions. Tactical operations centers support the maneuver of the combat elements, coordinate air strikes, and keep track of the battle. Commanders on the field viewing first hand the progress of the battle can be killed forcing new leaders to assume command. For both sides.

Throughout all of this, computers make no decisions about the outcome of warfighting. Computers execute the decisions of the warfighters involved, that is all. People do not fight computers here, people fight people.

CONVERGING TECHNOLOGIES

This advance in simulation is made possible by the very recent convergence of several technologies and innovative applications.

Computer Networking

First characterized by the ARPANET packet switching network, local area networking technology (LAN) has matured into off the shelf, standardized products. Packet switching protocols provide the means for transmitting data units needed by netted simulators and other gaming stations. Long haul networking (LHN) using wideband satellite or land lines, particularly the new capabilities being created with fiber optics, provide interfaces between LANs via gateways.

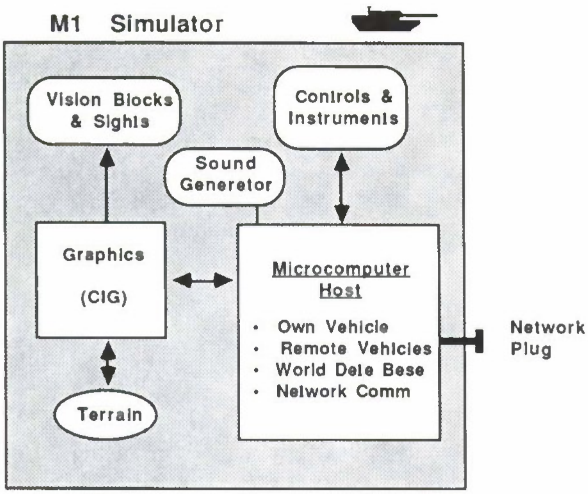
Communications

The communications capacity for running networks is expanding rapidly. C band wideband satellite capabilities are moving to Ku band with reduced cost and size. Fiber optic land lines, including those to Europe, are proliferating at a rapid rate. Whereas previously point to point connection schemes predominated, a variety of hybrid, reconfigurable schemes such as those featuring land lines that feed regional satellite uplinks broadcasting back to each site equipped with small receiver antennae are now discussed routinely. Self routing and self healing interconnections between sites are transparent to users.

Distributed Computing Architecture

There are many ways to structure computing resources on a LAN. The one that has worked the best so far is a completely distributed computing architecture where nearly all computing power resides in the simulators on the net. No mainframe or centralized computers are employed in an executive control or major computational role. As each simulator is plugged into the network, it brings the extra computing power needed to conduct network business given the network's larger size. No additional processors are needed.

Each simulator is a self-contained stand alone entity with its own host microprocessor, graphics, sound system, a complete copy of the terrain data base, and whatever else is needed to create a bubble of synthetic reality for its crew. This is similar to current simulator architectures except that each simulator host processor also has a fully functioning network



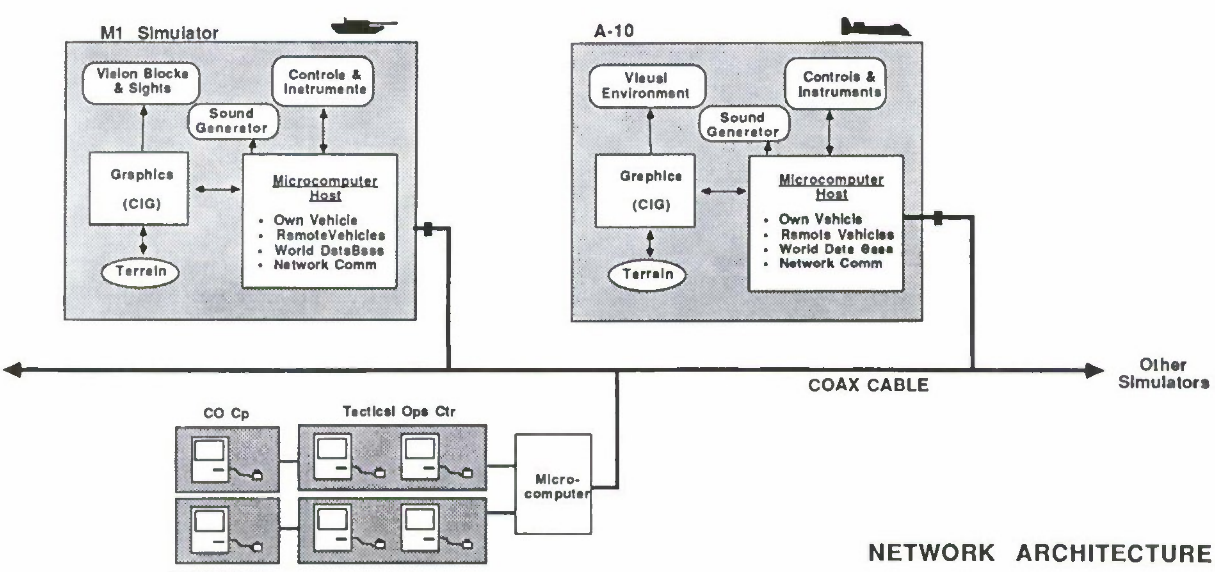
SINGLE SIMULATOR ARCHITECTURE

communications module which can transmit and receive messages. Simulators plug together via cable, transmitting and receiving data units from other simulators and gaming stations. When a simulator fails, the rest of the network continues minus the contributions of the failed device. Network degradations are soft and graceful.

Because each simulator is designed to be stand alone, specifically to be able to generate a complete set of cues to its crew without help from external processors, it can maintain a credible world for its crew should network transmissions suffer momentary interruptions. First, only a small amount of information is sent between simulators consisting mostly of orientation and position information (coordinates) and unique events ("I am simulator #16 and I have just hit simulator #22 with a round of SABOT in his left engine compartment." "I am simulator #22, I have just suffered a catastrophic kill, and I am now a burning hulk at coordinates ES89028876."). Second, each simulator is able to maintain predictive models about all other simulators on the network based upon the latest data packets from those simulators. If an update is slow in coming from another simulator, then its state can be inferred. When a new update is received, the actual state data is used in the next frame. If there is a serious discontinuity between the self generated inference and the newly received data message, algorithms can be activated to create a credible transition into the current state.

Network traffic using a distributed computing architecture turns out to be surprisingly modest. The need for message conflict resolution, the problem of senders and receivers of message packets all wanting to use the network at the same time, is minimized.

Also minimized is the problem of data corruption, another worrisome issue in networking. While every effort is made for pure data transmission, it is a lesser problem here for several reasons. There are relatively few message types and of these only a few are of such importance that they need acknowledgement by the recipient(s). Most messages are updates and are transitory: The next update is close behind. Since there is enough local processing power at each recipient to double check the



NETWORK ARCHITECTURE

credibility of an update, messages suspected of being bad can be discarded. Finally, if desired, network transactions can be allowed to mirror the dirtiness of chaotic operations in the real world: Ambiguity, error, and confusion are all properties of war and an occasional corrupted data element fits right in.

But perhaps the most important attribute of a distributed computing architecture is that it is an architecture for growth. Networks are the skeletons, microprocessors are the muscle, and communication protocols connect the two. As new, more powerful, cheaper microprocessors become available, they are simply plugged into the network. Outdated microprocessors are thrown away. As with the ARPANET, scores of different types of microprocessors using dissimilar operating systems can all talk via common network protocols.

This architecture allows anyone with a microprocessor to connect onto the net, either stand alone or as part of a simulator. A smart individual, armed with a microprocessor, can develop creative ideas off line and implement them on the net. The architecture is open and non-proprietary.

Simulator Design

There are many approaches to designing simulators, some which begin with physical models of the world and others which begin with behavioral or cue driven models of the world. In the first case, fidelity is defined by the match between the simulator's characteristics and a particular set of measurements from the physical world. In the second case, fidelity is defined by the strength and effectiveness of the cues which the simulator delivers to the operator, cues of specific information tailored to who the operator is and what he is doing in the simulator.

The attraction of the behavioral approach is that it can lead to the same results as the physical model but it is not held captive by it. Using the concept of selective fidelity, simulator and simulation characteristics which contribute directly to the goal of the training are represented in high fidelity, and those which do not contribute to the training are in low fidelity or not included at all.

Further, this approach recognizes the legitimacy of departing from the fidelity curve, including the use of exaggerations and fictions when they do not compromise the training goal. It also leads to the application of a rule taken from the discipline of industrial design: Do not make something appear to be what it isn't if broken expectations can be damaging.

Special Effects

Advances in sound synthesization, projection of infra sound, and application of design concepts from the special effects community have been used successfully to complement other traditional simulator cuing subsystems. Since simulations are illusions, the illusory technologies can enhance the end effect of a cue. Microprocessor based delivery systems make this affordable.

Graphics

The fast paced progress in microprocessors, integrated circuit design, and mathematical algorithms is nurturing advances in real time graphics. In almost all cases, the price for comparable performance is dropping dramatically. Further, the methods by which these machines render images are different than in the past making earlier measures of merit less valid. When coupled with selective fidelity cuing, the result is a new and powerful generation of graphic subsystems.

The use of selective fidelity has lead to an important new concept in graphics dealing with the relationship between environmental complexity in a scene and the display complexity used to present the scene. The predominate trend to date in computer image generation has been towards low environmental complexity (sparse data bases with minimal or no texture, few moving models, and modest correlation with specific topographic locations) and high display complexity (high screen resolution and high frame update rates). Data bases are costly to construct, few in number, difficult to modify, and machine specific. Cost of the CIGs has been high.

The graphics machine designed for warfighting in SIMNET reverses this trend. It is high in environmental complexity (many moving models and special effects, dense topological features with texture) and low in display complexity (relatively low numbers of pixels and an update rate of 15 frames per second for its eight channels). It is also very low in cost. The interesting functional phenomenon is that fighters viewing the combat world concentrate on its complexity, the only part that is tactically relevant, and adapt to its low resolution as they would a mud speckled window.

Rapid Prototyping R&D Process

Along with the evolution of these technologies comes a rigorous style of development characterized by what SIMNET calls the *60 % Solution* . This model recognizes the transient nature of any particular technology and the danger of freezing progress at a given technological plateau. It uses rapid prototyping to iterate on a specific technological solution but never tries to solve 100% of any problem at any time. This applies even if there is committee consensus about what the 100% solution is as articulated in a fully staffed and approved specification.

The 60% Solution closes on the goal, continually redefining and refining the objective, constructs prototypes and mock ups as interim by-products to verify direction, and cleans up the mess later. Best commercial practices replace mil-spec design philosophies.

The concept is that in a changing technological world, managing change is the *principal* role of the R&D process, not producing specific products. The 60% Solution is how SIMNET has responded to the Packard Commission recommendations.

HOW IS THIS DIFFERENT?

The inherent nature of networking gives rise to different ways to think about simulation. Some examples are below.

A Simulated World vs. a Single Simulator

Networking creates a simulated world. Combatants enter that world through their simulators or gaming stations, traverse that world, fight in that world, and are supported in that world by combat support and services support (e.g., refueling, rearming, and resupplying). Architecturally, like a piece of a hologram, as long as at least one microprocessor is living on the net and hosts a copy of the data base, the world lives....when other simulators join the network, their copies of the world are updated and their crews enter the current world.

As with other simulations, the simulated world is rent free, sustains no permanent ecological damage, and allows commanders to push their weapons, tactics, and organizations to the limit. The principal difference is the scale: Most simulators focus on the single crew. Networking creates a world of large forces.

This is a profound departure from simulation as we know it today. The foremost concern of every combatant is how to survive, fight, and win in this world. His simulator or gaming station is not an end in itself, inspected and certified for its micro-fidelity against a piece of hardware. The simulator is simply the entry device into the world, Alice's looking glass, and as long as it maintains a modest level of representativeness and does not perform in any obviously dumb manner, then the combatant takes it for what it is: The piece of equipment which he must adapt to in order to fight and win.

This is consistent with combat. Rarely does actual equipment perfectly match a manufacturer's engineering specification. Nor will weapon systems in combat perform like they do with good maintenance on sterile ranges with unstressed organizations in peacetime. An experienced commander knows this. He expects differences and organizes training programs to prepare for them. He understands that the hidden weapon in combat is the adaptable, creative, motivated man who can assess the characteristics of a combat world, determine what is needed to win, and make it happen with whatever hardware he can get his hands on, fix, modify, jury-rig, or whatever.

To date, the predominate thrust in simulation has been *inside* the single weapon system. Networking changes this. It creates a 24 hour, 7 days a week, "We Never Close" world where the predominate thrust is *outside* the single weapon system into the world of warfighting.

Experienced Teams vs. Novices

Because networking allows large teams to engage, the greatest benefit derives from the warfighting of combat personnel in operational units. These personnel have already developed their individual skills: Drivers know how to drive, pilots know how to fly, gunners know

how to engage and kill targets. Networking allows them to bring the warfighting team together and practice the integration of these skills.

This is not to suggest that lesser skilled students cannot benefit from being inserted into a combat world for training. The efficiency of simulation, coupled with the ability to tailor particular environments, makes this well suited for the institution. One can imagine recreating the great tactical battles of military history with students inserted into the battle interacting with the eventual outcome.

Rehearsal on a specific piece of terrain in the simulator might well raise the floor of proficiency for students so that subsequent exercises on that real terrain yields greater learning.

No Reset Button

In today's typical simulator session, the instructor usually initializes the simulator into a particular configuration and then conducts the training aimed at a given syllabus objective. Upon the conclusion of the session, or perhaps during the session, the instructor resets the simulator to various other conditions. This is efficient when training individual skills, but in continual combat operations where the crew is warfighting in a simulated world with other team mates, reset is a foreign concept.

When a combat vehicle runs low on gas, the crew must arrange to be refueled from a fuel truck, coordinating rendezvous, amount of fuel needed, protection from hostile forces, and rejoining the battle. When the supply in the fuel truck is too low to top off each vehicle, commanders must decide how they will modify their combat plans to accommodate this situation. The crew cannot just jump out of the simulator and press a reset button to get well again.

At first, this might seem to add inefficiency into training sessions. If these were traditional training sessions, then this would be so. When a crew goes off in the wrong direction during a maneuver, the tendency is to stop the maneuver, instruct the crew on their error, and begin again.

However, in the real world crews often make mistakes. It is part of the fog of war. Making mistakes and assessing and correcting these through the chain of command in the dynamics of warfighting is a rich form of trial and error learning. In combat, leadership includes being an assessor of performance and a remediator of the forces under command.

No Instructors, Controllers, or Umpires

In networked warfighting the combat team engages its opponent just as it would in the real world. This means that the chain of command on each side controls the battle to the best of its ability, issues operations orders, receives spot reports, maneuvers on the battlefield, and fights. Commanders trying to survey the battlefield can be killed and the chain must react and replace. Just as in combat, there are no overlords in this type of exercise other than the chain of command. None are necessary. Mentoring is the agent of improvement in leadership skills.

After Action Reviews Performed as in Combat

As above, the chain of command performs after action reviews as they would in combat. Even though networking allows for the collection of perfect knowledge about what each member of any conflict is doing at every moment of the battle, the only relevant information which the team requires is the same information it would have in combat. The combat model dominates training.

Real Time Casualty/Kill Removal

Just as commanders must pay greater attention to logistics, administrative, planning and execution factors when operating in a long term interactive world, there are similar concerns when crews who are injured or killed as a result of hostile action or accident must be immediately attended to or removed from the simulation. In both cases, the tactical situation can change drastically because of the reduction in force strength as well as the attendant burden of having to care for the injured, service damaged vehicles, call for personnel replacements, and insert them at the right time and place in the battle.

WHAT DOES THIS ALLOW US TO DO DIFFERENTLY?

Training - Fight the Present

Large scale simulator networking has obvious implications for training warfighting teams to a level of mastery never before seen outside of actual combat. Rapid train up of reserve forces, proficiency injections for new units or

those marginally capable, and Olympic training for those units already judged to be on top but which would like to go higher, are areas which might be possible with this new technology.

Future networks appear to be growable to sizes which could match the largest organizational structure, an attribute which is understood when one compares the similarity of the layered levels of combat organizations with nested networks. Just as the maneuver sectors of several battalions can be encompassed by the sector of an artillery battery, and several of these can be encompassed by the area covered by an aircraft, so too, one giant network or several nested interconnected networks can create the same world in simulation. If trends continue, it is likely that theater level exercises could be conducted in networked simulators in the mid term.

By 1988 R&D networks should be operational which can accommodate several hundred personnel. By 1990 that will expand to a few thousand. Commanders world wide, including Allied commanders, will have the ability to dial up training exercises to practice joint warfighting skills in a garrison setting.

This elevates simulator networking to a strategic level. It becomes a technology which offers new alternatives for the strategic positioning of U.S. forces world wide.

Configuration allows force-on-force training. Professional fighting forces compete vigorously when opposing each other. This is not true when fighting a computer. Video games become tiring. Networked combat derives its motivation from force-on-force ego invested competition. Coupled with the efficiency of large exercises conducted in networks, greater and greater garrison time can be involved in warfighting. Units can always be at war.

Networks can also provide access of the school house institutions to units for the delivery of new instructional material and technology, and in turn for feedback as to the effectiveness of training. Networks can serve many purposes in preparing our forces.

When the data base is of a crisis area and the order of battle reflects the latest intelligence, the coordination of team operations can be practiced in networked simulators. This is most critical. Many special situations in recent memory could have benefited from additional opportunity to practice teamwork under demand conditions. Since networks are easy to set up, it is possible to conduct dress rehearsal and contingency planning en route to the scene (e.g., shipboard) or nearby the crisis area.

Because combatants view their simulators as entry devices into a warfighting world, they have less tendency to distinguish between simulator failures and simulated failures. In both cases, their warfighting equipment and environment has been altered. In response, they adapt and fight on. This has implications for the rigor with which these simulators are maintained possibly resulting in cost savings.

The interesting by-product of networked exercises is that they exercise the chain of command in every respect. The chain of command must organize and supervise the use of networks as well as the warfighting that goes on inside of them. Leaders are trained at every juncture and practice what they have learned.

Development - Fight the Future

Team simulation introduces a new tool for the development of weapon systems. This is made possible because the simulators can be employed in simulated combat with the same force size and tactics expected of the candidate system against base line systems (other networked simulators) representing the expected threat and manned by aggressors trained in the tactics of the opponent. This expands and complements the design data collected in engineering simulators at Service and contractor R&D centers.

Industrial and university centers can have small networks on which to do their research and development. As ideas and designs mature, these laboratories can be netted to the larger world for test. Challenge matches can be arranged and designs tested in the caldron of battle.

When typical troops are used in this context, training and tactics have to be addressed early in the development cycle. Prototype training systems must be developed to prepare the troops to use the candidate systems. Potential problems in training, human factors, manning, organization, and implementable tactics are discovered early on.

For those good ideas and designs which rise to the top, the developer has a rich environment in which to show them off: The audience dons combat gear, enters the simulated world, and warfights in the candidate weapon system against the threat. Instead of communicating

with thick proposals or lengthy briefings, government officials and legislators can live the weapon system in combat conditions.

If it is decided to go forth with full scale development of the weapon system, the training subsystem, including the training simulators, are already developed. The training system can be fielded before the fielding of the weapon system as it should but rarely is.

This use of the networked world as a theater applies to demonstrations of U.S. weapons which are being considered for foreign military sales as well.

Testing and Cost Projections

The testing requirements for new weapon systems are rigorous but often can only be accomplished under restrictive conditions, e.g., safety constraints that limit realistic maneuver, use a small number of early test vehicles unrepresentative of actual employment strength, and do not employ the system as a fightable weapon adapted by its operators to changing conditions to maximize strengths. On the other hand, team simulation is not limited by these constraints and can complement testing by providing data in these areas.

Similarly, cost projections of life cycle costs often make many assumptions about how typical forces will use the system. Data from interactive simulations where typical combatants fight the candidate systems against base line forces can augment cost models.

Future Command and Control Systems

One far reaching and less obvious attribute of simulator networking is that it is a mimic of future command and control structures. A joint AIRLAND battle of multi-battalion size with air, land, command, and support elements networked between several sites by long haul networking is, in effect, a real time, sophisticated command and control system. As the simulator networking technology is developed to allow this level of exercising, there is a direct advance in the state of command and control systems.

In the same sense, networks that span Allied forces for NATO exercises are at the heart of interoperability. Networks which can be successfully constructed across these boundaries will aid in the solution of interoperability issues.

WHERE ARE WE GOING?

Ongoing R&D on large scale simulator networking will have several influences on the course of simulation. Some likely trends are suggested below.

International Nets

The international networking infrastructure for world wide simulator networks will grow over the next several years. Digital communication capacity will support this at low cost. Networks will be connected for large exercises when needed, or be operated in smaller clusters for local use.

Networkable Simulators

To get the maximum use of these networks, DoD will likely require that all simulators procured in the future are network capable. The importance of mastering warfighting is a foremost military need, and this is an important step to provide this capability. All networks will be standardized by DoD and will be interconnectable.

Common Cues

The major technical issue will be how to construct functionally equivalent data bases, specifically the equivalence in cues provided to crews operating in the same world but in different types/manufacturers' simulators. This will be aggravated by the pace of technology. We can expect to see many different generations and types of simulators residing on a given network just as many different styles and ages of telephones are plugged into the telephone system. The increased capability of newer simulators must be compatible with the capabilities of earlier generation machines. A newer simulator cannot be allowed to give its crew an artificial, unearned, unfair tactical advantage.

Affordability

Because of the numbers of simulators which will be needed for large team practice across the U.S. and NATO networks, the unit cost of new simulators must be dramatically lower than simulators today. Work ongoing in this area has demonstrated that this is possible and in fact is desirable given the pace of the core technologies. With technology moving so quickly, investing heavily at a given technology level has severe penalties. The concept of an objective, finished system ready for long term

procurement belies the technological realities of today's world. Low cost systems which have paid for themselves and can be removed from service after five years capitalize on technological advancement and keep simulation on the cutting edge. This argues for a process based upon best commercial practices (non-mil-spec) and a very restrained logistic infrastructure.

Look To The Outside

The dominate orientation for simulator designers should be to the warfighting world outside the simulator, not inside. For those life or death battles in which the combatant has fully projected himself, the effort and money that goes into micro-fidelity has little return. It is the interaction of the individual and his crew with the world outside which deserves the highest attention to fidelity.

60% Solution

All of the above argues for an R&D approach which uses the 60% Solution: Develop quickly, be satisfied with good enough, keep the development cost and recurring costs low, and plan to throw away earlier than in the past. To keep pace with this approach, the requirements process from Service users will have to allow rapid, iterative development and fielding of less than perfect devices. In the end, however, this process will provide superior solutions to the user for less money.

ABOUT THE AUTHOR

Lt Col Thorpe is the DARPA program manager for advanced research projects in simulation and training. He manages the SIMNET Large Scale Simulator Networking Program. His Ph.D. is in Industrial Psychology from Bowling Green State University, Ohio. He is a graduate of the Squadron Officer School and graduated with highest distinction from the College of Naval Command and Staff, Naval War College.

REFERENCES

1. Thorpe, Jack A., Captain, USAF. Future Views: Aircrew Training 1980-2000, Air Force Office of Scientific Research, 15 September 1978, unpublished concept paper. Available from author.
2. Vuono, Carl E., General, USA, Chief of Staff. Armor Conference 1987 - Welcome and Keynote Address, Ft. Knox. Kentucky.

References (Con't)
3. Pope, Arthur R. ; Langevin, Tim; Tosswill, Andrew R. The SIMNET Management, Command and Control System, Report No. 6473, March 1987, prepared by BBN Labs, 10 Moulton Street, Cambridge, MA 02238.
4. Pope, Arthur R. The SIMNET Network and Protocol, Report No. 6369, February1987, prepared by BBN Labs, 10 Moulton Street, Cambridge, MA 02238.
5. Herman, Jane. A New Approach to Collective Training Simulation: The SIMNET Simulation Formula for Success, Technical Report, January 1987, prepared by Perceptronics, Training and Simulation Division, 21122 Erwin Street, Woodland Hills, CA 91367.
6. Chung, James W.; Dickens, Alan R.; O'Toole, Brian P.; Chiang, Carol J. SIMNET M1 Abrams Main Battle Tank Simulation, Report No 6323, January 1987, prepared by BBN Labs, 10 Moulton Street, Cambridge, MA 02238.
7. Cyrus, Michael L. SIMNET Computer Image Generation System. BBN Delta Graphics Inc.
8. Packard Commission Report: Quest for Excellence, Final Report, June 1986.
9. Gorman, Paul. F., General, USA (Retired), President, Cardinal Point. Advice to the Services, Presented at the Eighth Interservice Industry Training Systems Conference, November 1986.

SMARTER LOGISTICS TO MEET THE CHALLENGE OF CLS AND COMS

Brian J. Williams
Burtek, Inc.
Tulsa, Oklahoma

ABSTRACT

Contractor Logistic Support and Contractor Operation and Maintenance Support contracts are becoming widely used for the upkeep and maintenance of Government Training Systems, replacing traditional organic support maintenance methods. Users are benefitting from this transition because of the increase in trainer availability that has resulted; but it has placed an extra responsibility on the manufacturer to provide the required support at a competitive cost. The cost of support has become an evaluation factor in the procurement process, making it essential to properly evaluate and control all the factors that influence support costs from conceptual design to the development of the maintenance concept. This paper discusses the role logistics plays in design and how it is responding to the challenge of contractor supported programs, including the development and greater use of computer programs, and the influence these tools should play in future training system procurement programs.

INTRODUCTION

Over the last few years there has been increasing concern over the proportion of the defense budget spent on the support of in-service equipment and on the manpower resources this equipment absorbs for maintenance. This has been particularly true in the case of military training systems where the Government has mandated that Contractor Logistic Support (CLS) or Commercial Operation, Maintenance and Support (COMS) replace traditional organic support methods. This move is proving to be successful for the Government and user in terms of lower support costs and greater system availability, but at the same time, it has placed a greater responsibility on training system suppliers, and on the logistician in particular, for it is often the cost and effectiveness of the CLS or COMS portion of their bids that determines their overall success in winning new business. Innovative and cost-reducing designs and support concepts have to be developed in order to survive in what has become a highly competitive marketplace. Logistics has had to become smarter, not only to provide the best possible user support, but also to identify, evaluate and control those factors that influence the in-service support costs. For these reasons, logistic computer tools are becoming more widely used and accepted by the logistics community and others in industry and Government procurement agencies.

IMPACT OF CLS

Users are enjoying a growing capability in their training systems as more powerful computers become available and as more realistic motion and visual systems are developed. This has given the user more true-to-type and effective training devices, but has also given his maintenance organization an increased burden due to the extra sophistication and complication of the new equipment. This trend is happening at a time of severe manpower restrictions in the U.S. military, and has led to the policy of phasing out organic support maintenance (by enlisted personnel) in favor of CLS or COMS.

The move to CLS has given rise to great improvements in training system availability since its introduction in the early 1980's. Some reports suggest that availability in some cases has increased from around 60% to well over 90%, due in part to the efficiency of the contractor reacting to the commercial pressure to maintain customer satisfaction and increase effectivity, while minimizing equipment downtime and repair costs. CLS contracts often include stringent penalty payments which are imposed if the required standards of availability are not achieved. The contractor must therefore be capable of making accurate and timely logistic evaluations throughout the program, especially during equipment design, development, and contract proposal phases. Since CLS costs are usually evaluated at the same time as the initial acquisition costs for the "hardware," the contractor is committed to properly evaluate and minimize these costs in order to stay competitive and profitable. If support is not properly considered, then the results can be serious for the contractor. Even simple problems in the field can cost hundreds of times the cost of rectifying the problem in the factory if only it had been identified and corrected there. Under organic support, this would not have concerned the manufacturer too much since he was largely insulated from the inherent cost of maintenance and support beyond the warranty period. CLS and COMS now makes the manufacturer more responsive to these problems since the costs involved now impinge directly on profitability.

In recognition of this situation, the role of logistic planning and assessment in design assumes greater importance.

LOGISTICS ROLE IN DESIGN

Many manufacturers will confess to being able to survive in the marketplace producing satisfactory products without any defined logistic organization or without devoting resources during design to optimizing logistic support. It is very likely, however, that such equipment would either have unsatisfactory reliability and maintainability performance in the field, or would have unnecessarily high support costs. Users of such equipment will recognize the many symptoms of poor logistics planning and design such as insufficient spares, lack of test equipment, poor maintainability, and so on.

To overcome these problems, reliability, maintainability, supportability, and Life Cycle Cost (LCC) analyses must be fully integrated into the design process in order to assess proposed designs for supportability, identify problem areas, and to quantify resources necessary to provide the required level of support at the minimum cost. Studies have shown that, to be effective, the logistic planning effort must be made at the earliest opportunity in the system life cycle. As Figure 1 shows, around 90% of the potential LCC of a product is already committed by the end of the development stage, although relatively little (less than 50%) has been actually spent by this time. That is to say, if logistic input is not made early in the design process, then it will come too late to make any significant difference to the overall cost of ownership of that system. Further, many studies have shown that the in-service costs account for over half of the total life cycle cost of the system, acquisition costs sometimes being as little as a quarter or a third of the total costs, as illustrated in Figure 2. The role of logistics in design is therefore, to identify, evaluate, and control the "below the line" costs represented by the "iceberg" diagram shown in Figure 3, - the costs that could sink the "program ship" if not properly evaluated during design.

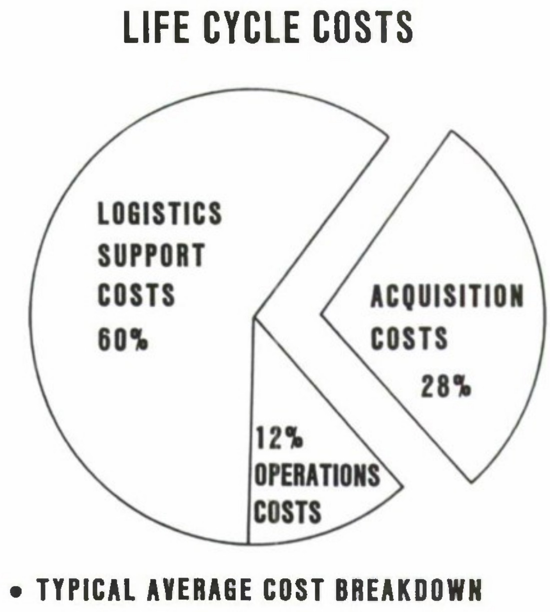


Figure 2. Typical Average Cost Breakdown

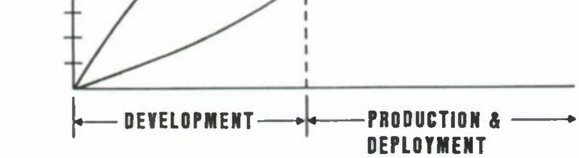


Figure 1. Life Cycle Cost Profile

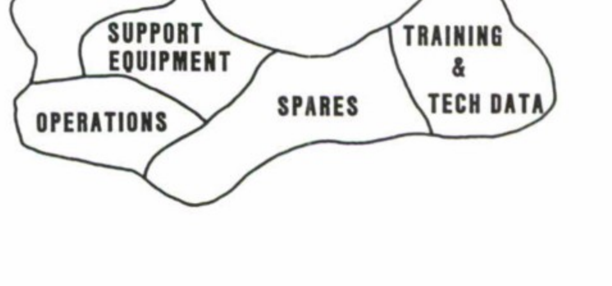


Figure 3. Total Cost Visibility

HOW LOGISTICS CAN MEET THE CHALLENGE OF CLS

Having established the proper role of logistics in design and in developing cost effective maintenance concepts, the cost drivers that influence the effectiveness of the CLS effort must be identified and evaluated. The problem is complex because of the numerous variables in the support equation. Use of computers are, of course the solution to this complex problem, and software is available to assist the logistician. Traditionally, much of this software has only been available for use on large main-frame computers requiring detailed (and costly) data preparation, suitable for major programs with long development times and adequate resources, but less suited to the training system world, where time and resources to carry out logistic studies is often very restricted. Fortunately, the increased availability of software for the desk top PC in recent years has made these tools viable for use on even the smallest program. For example, computer models are available that automate and greatly simplify the reliability prediction process enabling designs to be evaluated and trade offs made very early in the design process. Logistic Support Analysis (LSA) computer models have been developed that automate the integration of logistics into design, available in PC format for data input and output, although requiring interface to a main frame computer for the data processing required; and Life Cycle Cost models have been transferred from the large, expensive mainframe computers to the PC facilitating rapid and cost effective analysis of support requirements. With the tools available, the logistician can optimize reliability and cost of an item, for example, by varying the quality standards of the procured components to meet reliability and cost goals, then apply an LCC model to optimize the maintenance concept, spares, support equipment, manpower and other resources, iterating the process as necessary to meet a minimum LCC. This process can highlight where, for example, increased acquisition costs can be justified by a reduction in the support costs over the life of a system. Accounting methods such as inflation and discount rates can also be included in the models to determine their effects on the resultant LCC.

Thus the logistician does have the capability to make the critical design and support decisions that are required for the success of the CLS phase. The tools available enable a better analysis of all trade-off decisions, allowing quantative assessments to be made for any cost or availability requirements, as demonstrated by the following example, which shows one application of a LCC model.

LCC CASE STUDY

For this example, Burtek has used the USAF LCC program CASA (Cost Analysis and Strategy Assessment), developed by the Defense Systems Management College[1]. This model is particularly suitable for most training equipment applications because of the relatively small size of the model required and the convenience and cost advantages of the PC use. Production and start-up costs are used as input to the model, together with details of the system configuration, maintenance policy, reliability and maintainability data, spares costs, training costs, and other acquisition and support data. Life cycle cost data is produced in a variety of formats depending on the analyst's requirements. For example, support equipment utilization can be shown to determine what equipment is essential with loading factors; maintenance personnel requirements, yearly support cost tables, LRU spares and repair parts costs, availability, and other valuable data can also be calculated. The model can also compute, for a given maintenance policy, spares requirements optimized to meet cost, system availability, and stock out risk criteria. By performing various simulations, it is possible to vary different parameters, such as the maintenance policy to compare different results, evaluate risks, and to identify the best design and support option.

Consider a training simulator comprising a computational system, I/O Interface System, Image Generation System, Power Distribution System, and a Student Station (cockpit), as shown in Figure 4. Four systems are to be deployed at one site. Support is to be by CLS under a 15 year contract. Initial logistic studies have determined the optimum design of the system in terms of equipment selection. Trade-off studies between possible computer configurations have identified the best solution regarding performance, LCC, and reliability. At issue now is the support philosophy for the trainer, that is, whether it is better to give the CLS staff the capability (training and test equipment) to enable on-site repair of faulty LRU's, or to return the items to factory (depot) for repair. Studies have determined that all but the computer system will be repaired on-site as the available manpower and resources are adequate to carry out the level of repair required. To repair the computer, additional comprehensive test facilities are required, costing around $125,000.00, plus additional technician training. The question is whether this additional "up front" cost could be justified over the life of the equipment.

The first reaction from the cost management organization would probably be to reject the decision to procure the extra facilities because of the costs involved, and there would be great pressure on the logistic organization not to pursue this course of action (particularly since many procurement agencies do not look much farther beyond the acquisition phase and the first few years of CLS). However, detailed studies using CASA highlights that, over the long support period of 15 years, the least costly approach would be to buy the test equipment and repair the computer

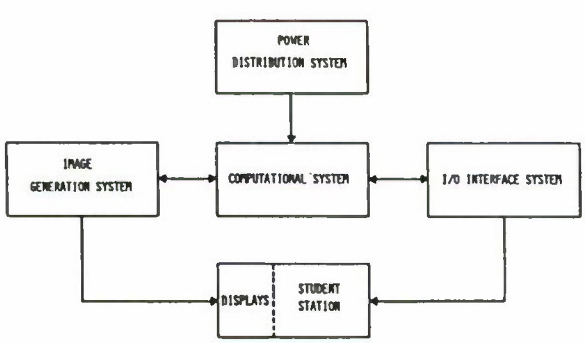


Figure 4. Training System Block Diagram

LRU's on-site. Although this is an extreme example, it does demonstrate the usefulness of allocating time and resources early in the design phase or proposal phase to make these assessments, and the importance to procurement agencies of considering long term implications of support decisions.

Figures 5 - 7 highlight the output results from CASA (edited for clarity), showing the differences in costs for the two support options. Figure 5 compares the acquisition costs for on-site repair of computer LRU and depot repair of computer LRU's. Figure 6 compares the operation and support costs for the two options, while Figure 7 compares the spares and trainer operational availability for the two options.

The major differences between the two options are summarized below.

a. <u>Acquisition Costs, Support Equipment Maintenance and Training</u>. These costs are obviously more for on-site repair due to the extra support equipment required, although the difference is offset by the reduction in LRU spares. In this example, rather simplistically, the system MTBF and the on-site repair time of each LRU is such that the availability target can be achieved with no spares if the computer LRUs are repaired on-site.

b. <u>Repair Parts and Material</u>. Although on-site costs are greater if LRU's are repaired there, the cost of repairs at depot is considerably more. This is because repairing at depot/factory incurs the full overhead for repair labor and shipping costs, whereas on-site repair only incurs the cost of material consumed (typically electronic parts such as I/C's or resistors). Labor rates on-site are effectively zero for the repair activity since manpower already exists and is paid for under the CLS budget. Past experience and CASA studies show that the manpower requirements for CLS are determined more by such factors as minimum levels required by safety regulations and shift working rather than the amount of maintenance required (preventive and on -

```
        COST ANALYSIS AND STRATEGY ASSESSMENT (CASA) MODEL--VERSION 1.0
       ****************************************************************

                                  DEFENSE SYSTEMS
TRAINER - ON-SITE REPAIR        MANAGEMENT COLLEGE

                                 ACQUISITION COSTS
                                 *****************

PRODUCTION COSTS  - Not Included                                       0.

SYSTEM ACQUISITION - Not Included                                      0.

SUPPORT EQUIPMENT               QTY   COST/UNIT
(ORGANIZATIONAL)
    OSCILLOSCOPE                 1       5000.       5000.
    MULTIMETER                   1        400.        400.
    CARD TESTER                  1      45000.      45000.
    DOT BAR GENERATOR            1       2000.       2000.
    VME CHASIS                   1      25000.      25000.
    LOGIC ANYLIZER               1      15000.      15000.
    COMPUTER TEST EQUIP.         1     125000.     125000.

                                                                  217400.
SPARES
(ORGANIZATIONAL)


        NO LRU SPARES REQUIRED
                                                                       0.

TECHNICAL DATA - DEVELOPMENT COST                                  84542.

TRAINING - DEVELOPMENT COST                               166332.

  TRAINEE COST                                             63160.

  TRAINING DEVICES             QTY   COST/UNIT
    VIDEO VIEWING EQ.            1       1000.       1000.
                                                                  230492.

ITEM MANAGEMENT                                                     5000.


MISCELLANEOUS ACQUISITION COSTS                                      708.
                                                                 -------
TOTAL ACQUISITION COST                                            552142.
                                                                 -------
```

```
        COST ANALYSIS AND STRATEGY ASSESSMENT (CASA) MODEL--VERSION 1.0
       ****************************************************************

                                  DEFENSE SYSTEMS
TRAINER - DEPOT REPAIR          MANAGEMENT COLLEGE

                                 ACQUISITION COSTS
                                 *****************
PRODUCTION COSTS  - Not Included                                       0.

SYSTEM ACQUISITION - Not Included                                      0.

SUPPORT EQUIPMENT               QTY   COST/UNIT
(ORGANIZATIONAL)
    OSCILLOSCOPE                 1       5000.       5000.
    MULTIMETER                   1        400.        400.
    CARD TESTER                  1      45000.      45000.
    DOT BAR GENERATOR            1       2000.       2000.      52400.

SPARES                          QTY   COST/UNIT
(ORGANIZATIONAL-COMPUTER ONLY)
    POWER SUPPLY SUB ASS         1       5000.       5000.
    CASHE SEL SUB                1       5238.       5238.
    MICRO SEQ.                   1       7845.       7845.
    I/E  II                      1       6335.       6335.
    2MB MEMORY                   1      15000.      15000.
    I/O MICRO PROCESSOR          1        523.        523.
    POWER SUPPLY                 1        800.        800.
    MBD ASSEMBLY                 1       5000.       5000.
    DISC DRIVE                   1       6375.       6375.
    TAPE PROCESSOR               1       3590.       3590.
    TAPE DRIVE                   2       5000.      10000.
    CRT                          2        695.       1390.
    PRINTER                      1        500.        500.
    PDU                          1       6368.       6368.
    POWER SUPPLY                 1       3188.       3188.
    MICRO PROCESSOR              1       3585.       3585.
    MASS STORAGE MODULE          2       3250.       6500.
    GLOBAL MEMORY                1       1185.       1185.
    TOTAL (ALL SPARES)                                         170461.

TECHNICAL DATA - DEVELOPMENT COST                               84542.

TRAINING - DEVELOPMENT COST                          104766.

  TRAINEE COST                                        26940.

  TRAINING DEVICES             QTY   COST/UNIT
    VIDEO VIEWING EQ.            1       1000.       1000.     132706.

ITEM MANAGEMENT                                                  5000.

MISCELLANEOUS ACQUISITION COSTS                                   708.
                                                              -------
TOTAL ACQUISITION COST                                         445817.
                                                              -------
```

Figure 5. Comparison of Acquisition Costs

TRAINER - ON-SITE REPAIR

OPERATION AND SUPPORT COSTS

	ORGANIZATIONAL	DEPOT	TOTAL
SUPPORT EQUIP MAINT	33697.	0.	33697.
RECURRING TRAINING	600.	0.	600.
REPAIR PARTS AND MTL	28006.	0.	28006.
CONSUMABLES	28.	0.	28.
CONDEMNATION SPARES	5767.	0.	5767.
TECH DATA REVISIONS	11625.	0.	11625.
TRANSPORTATION	514.	0.	514.
RECURRING ITEM MGNT	2945.	0.	2945.
CONTRACTOR SERVICES	1260000.	0.	1260000.
MISC O & S COSTS	6920.	0.	6920.
TOTAL O & S COST	1350103.	0.	1350103.

TOTAL LIFE CYCLE COST FOR 192 MONTHS........ 1902245.

TRAINER - DEPOT REPAIR

OPERATION AND SUPPORT COSTS

	ORGANIZATIONAL	DEPOT	TOTAL
SUPPORT EQUIP MAINT	8122.	0.	8122.
RECURRING TRAINING	440.	0.	440.
REPAIR PARTS AND MTL	8419.	2213223.	2221642.
CONSUMABLES	8.	2213.	2222.
CONDEMNATION SPARES	169.	5596.	5765.
TECH DATA REVISIONS	11625.	0.	11625.
TRANSPORTATION	33.	48444.	48476.
RECURRING ITEM MGNT	13950.	0.	13950.
CONTRACTOR SERVICES	1260000.	0.	1260000.
MISC O & S COSTS	354.	0.	354.
TOTAL O & S COST	1303120.	2269475.	3572596.

TOTAL LIFE CYCLE COST FOR 192 MONTHS........ 4032413.

Figure 6. Comparison Of Operation And Support Costs

ON - SITE REPAIR

OPERATIONAL AVAILABILITY ANALYSIS

WITH LRU SPARES QUANTITIES DETERMINED BY OPTIMIZATION

(COMPUTER SPARES ONLY SHOWN)

LRU NAME	UNIT COST	OPS	MTBF	TAT (MOS)	SPARES/ LOC.	LOGIS. AVAIL.
POWER SUPPLY BUS ASS	5000.	1	55556.	.021	0	.99988
BUS TERMINATOR DECOD	961.	1	374532.	.021	0	.99998
CLOCK	1478.	1	273973.	.021	0	.99998
IOP BUS TERMINATOR	437.	1	1298701.	.021	0	.99999
I/O ASSEMBLY	2440.	1	115741.	.021	0	.99994
CASHE SEL BUS	9239.	1	50176.	.021	0	.99987
MICRO SEQ.	7945.	1	11174.	.020	0	.99942
I/E 11	6335.	1	27624.	.021	0	.99976
PMS CONTROL	5000.	1	413223.	.021	0	.99998
PARALLEL GRAPHICS	764.	1	61805.	.021	0	.99989
2MB MEMORY	15000.	1	15232.	.021	0	.99956
I/O MICRO PROCESSOR	923.	1	22442.	.021	0	.99970
FOREPLANE	263.	1	4166667.	.021	0	1.00000
BACKPLANE	387.	1	200000.	.020	0	.99997
BACKPLANE CONN.	310.	2	2000000.	.021	0	.99999
8 LINE ASYNC CONTROL	2500.	1	98522.	.021	0	.99993
DISTRIBUTION PANNEL	700.	1	2000000.	.021	0	1.00000
POWER SUPPLY	900.	1	40000.	.021	0	.99983
HSD ASSEMBLY	5000.	2	71582.	.021	0	.99981
RTOM	2000.	1	134424.	.021	0	.99995
DISC DI	5472.	1	127714.	.021	0	.99995
UNIV. DISC PROSSOR	8703.	1	68681.	.021	0	.99990
DISC DRIVE	6375.	1	8000.	.021	0	.99916
TAPE PROCESSOR	3890.	1	21997.	.021	0	.99969
TAPE DRIVE	5000.	1	5500.	.021	0	.99877
CRT	695.	3	10000.	.021	0	.99798
PRINTER	500.	1	40000.	.021	0	.99983
PDU	6369.	1	24528.	.021	0	.99972
TOTAL TRAINER	0.	1	326.	.024	0	.97695

LRU SPARES COST PER LOCATION = 0.

SYSTEM LOGISTICS AVAILABILITY = .97695

SYSTEM INHERENT AVAILABILITY = .99465

SYSTEM OPERATIONAL AVAILABILITY = .97184

DEPOT REPAIR

OPERATIONAL AVAILABILITY ANALYSIS

WITH LRU SPARES QUANTITIES DETERMINED BY OPTIMIZATION

(COMPUTER SPARES ONLY SHOWN)

LRU NAME	UNIT COST	OPS	MTBF	TAT (MOS)	SPARES/ LOC.	LOGIS. AVAIL.
POWER SUPPLY BUS ASS	5000.	1	55556.	1.000	1	.99993
BUS TERMINATOR DECOD	961.	1	374532.	1.000	0	.99914
CLOCK	1478.	1	273973.	1.000	0	.99882
IOP BUS TERMINATOR	437.	1	1298701.	1.000	1	1.00000
I/O ASSEMBLY	2440.	1	115741.	1.000	1	.99998
CASHE SEL BUS	9239.	1	50176.	1.000	0	.99360
MICRO SEQ.	7945.	1	11174.	1.000	1	.99839
I/E 11	6335.	1	27624.	1.000	1	.99973
PMS CONTROL	5000.	1	413223.	1.000	0	.99922
PARALLEL GRAPHICS	764.	1	61805.	1.000	1	.99995
2MB MEMORY	15000.	1	15232.	1.000	1	.99913
I/O MICRO PROCESSOR	923.	1	22442.	1.000	1	.99959
FOREPLANE	263.	1	4166667.	1.000	1	1.00000
BACKPLANE	387.	1	200000.	1.000	1	.99999
BACKPLANE CONN.	310.	2	2000000.	1.000	1	1.00000
8 LINE ASYNC CONTROL	2500.	1	98522.	1.000	1	.99998
DISTRIBUTION PANNEL	700.	1	2000000.	1.000	0	.99984
POWER SUPPLY	900.	1	40000.	1.000	1	.99987
HSD ASSEMBLY	5000.	2	71582.	1.000	1	.99984
RTOM	2000.	1	134424.	1.000	1	.99999
DISC DI	5472.	1	127714.	1.000	0	.99748
UNIV. DISC PROSSOR	8703.	1	68681.	1.000	0	.99532
DISC DRIVE	6375.	1	8000.	1.000	1	.99691
TAPE PROCESSOR	3890.	1	21997.	1.000	1	.99958
TAPE DRIVE	5000.	1	5500.	1.000	2	.99952
CRT	695.	3	10000.	1.000	3	.99981
PRINTER	500.	1	40000.	1.000	1	.99987
PDU	6369.	1	24528.	1.000	1	.99966
TOTAL TRAINER	0.	1	326.	.050	0	.95310

LRU SPARES COST PER LOCATION = 170433.

SYSTEM LOGISTICS AVAILABILITY = .95310

SYSTEM INHERENT AVAILABILITY = .99933

SYSTEM OPERATIONAL AVAILABILITY = .95249

Figure 7. Comparison of Availabilities

equipment corrective). Indeed many simulator installations could be adequately supported by predominantly on-call maintenance personnel if this approach was acceptable to the user. If traditional manning levels are required, primarily for insurance purposes, then large amounts of "dead time" or waiting time will result while the maintainer waits for unscheduled maintenance actions to occur. In this event, effective and efficient use of available manpower could be used for carrying out on-site repair of LRUs. This results in large cost savings for repair as shown by CASA. Over a long support period the cost savings become very significant, enough to more than offset the initial additional acquisition costs.

c. Availability. A minimum system operational availability of 95% was required. Both options achieve this requirement, although interestingly enough, the option utilizing on-site repair achieves a better availability with no spare LRU's because of the relatively few failures predicted per month. (The minimum required availability can be achieved providing the failed LRU is repaired within one working day on site.)

SUMMARY

The complexity and sophistication of today's training systems, coupled with the stringent availability requirements placed on CLS or COMS contractors in a highly cost-conscious marketplace, places increased importance on the timely availability of accurate logistic data to aid the design and support development decision making process. The ability of the system to meet the users' training requirements, and the profitability and effectiveness of the CLS operation, depends largely on the success of the contractor's logistic operation to respond to the challenge. Logistics tools to assist in this requirement exist and are beginning to be widely used within the logistics community. They are demonstrating useful results and should continue to play an increasingly important role in the future in the procurement process for new training systems.

REFERENCE

[1] Cost Analysis and Strategy Assessment (CASA) models and documentation Defense Systems Management College, DRI-PMSS, Fort Belvoir, Virginia.

ABOUT THE AUTHOR

Mr. Brian Williams is the Supervisor, Engineering Services, within the Integrated Logistic Support (ILS) Department of Burtek. He is currently responsible for providing engineering support to design for reliability, maintainability, safety and human factors for Burtek's training systems. He is also responsible for the Life Cycle Cost and Logistic Support Analysis and Provisioning functions of ILS. He holds a Bachelor of Science Degree in Engineering from the University of Bath, England, is a member of the Institution of Mechanical Engineers, and a member of the Society of Logistics Engineers (SOLE). He was formerly employed as Logistics Section Head at British Aerospace, Brachnell, England, responsible for all integrated logistic support activity within that company's naval equipment division. In earlier associations, he was a principal support engineer with Sperry Gyroscope; Material Manager with the British Aerospace field logistics team working with a Middle East Air Force; and section leader responsible for logistics at British Aerospace, Bristol, England.

C-130 AIRCREW TRAINING SYSTEM (ATS) ACQUISITION: USING AND SUPPORTING COMMAND'S LESSONS LEARNED (ASSURING THE CRITICAL ADVANTAGE)

Lt Col Ronald E. Dukes
Chief of ISD & Training Systems Technology
Directorate of Aircrew Training & Resource Management
Headquarters Military Airlift Command
Scott AFB, Illinois

Ms Jo D. Voeller
Logistics Management Specialist
Training Devices System Management Division
Ogden Air Logistics Center
Hill AFB, Utah

ABSTRACT

As the number and complexity of training programs converting to total training systems increases, the role of the using and supporting commands in the source selection process is becoming larger and more vital than ever. This expanded role is required as each training program has its own associated training objectives and integration requirements whose expertise resides in the operational and logistics management arena. This is different than previous acquisitions where only equipment was being procured and the required expertise resided in the engineering arena within the procurement agency.

INTRODUCTION

The US Air Force, and in particular, the Military Airlift Command, is currently undergoing a revolution in aircrew training. Training technology, new design approaches, new development techniques, and new instructional strategies are combining to bring us out of the lock step, chalkboard, 35mm slide arena that has been part of our training programs for the last 20 years. Current state-of-the-art training systems acquisitions are de-emphasizing detailed hardware specifications as significant drivers in procurement. Hardware requirements are rightfully becoming a part of the media analysis process which is done in conjunction with program development.

New training systems emphasize tasks, objectives, and performance standards in defining the training requirements. Hardware requirements are then derived as part of the formal ISD process. Guaranteed student throughput and cost per student have become the important considerations with this approach. Ultimately, the final flight evaluation is the vehicle which ensures the requirements of the training systems are met.

C-130 ATS DEFINITION

We believe it is important to define the basic concepts of C-130 ATS in order to better appreciate lessons learned from the acquisition phase of the program.

The C-130 ATS evolved from an initial concept of a contractor developed, Air Force conducted program into a totally contracted training system concept. The C-130 ATS contract includes 28 courses for the DOD C-130 formal school and all Military Airlift Command C-130E and H model continuation training, including tactical. The system includes optimized use of existing training assets, including ten weapon system trainers, two cockpit procedures trainers, and numerous part task trainers which are furnished to the contractor by the government, as is. It also includes all maintenance and logistic support for the weapon system trainers, cockpit procedures trainers, and other part task trainers within the program. It includes total system management of all ground based training using automated management tools, all scheduling, and all training scenarios for the flying environment. It also includes a proficiency-based training continuum which begins with entry into the formal school and ends either in transfer out of the weapon system or retirement.

CONTRACTOR RESPONSIBILITIES

Under the C-130 ATS concept the contractor will be responsible for the entire ISD process from beginning to end, including formative, summative, and operational evaluation. They will be responsible for development and production of all courseware, all ground instruction, all hardware modifications and any new software development. They will also be responsible for total operation, maintenance, and support of the ground based training system; all student management; administration; configuration management; and quality assurance.

AIR FORCE RESPONSIBILITIES

The Air Force was responsible for the initial development of training requirements which included detailed tasks, objectives, and performance standards. This was accomplished in an 18 month front-end analysis effort known as the Model Aircrew Training System and was completed prior to RFP development. The Air Force is currently responsible for all facilities, turnover of existing assets, flight instruction, flight evaluations, and providing subject matter experts during the development phase of the program. Additionally, the Air Force is responsible for providing feedback into the system, active oversight of the contract including maintaining a recompetition package, and quality assurance of both operational and logistics activities.

LESSONS LEARNED

The lessons learned for this program will be presented chronologically in the same order as they occurred in the development and acquisition process.

The C-130 ATS began with a front-end analysis alluded to earlier. This analysis included a system baseline analysis and a training requirements analysis which provided the basic concepts from which the program was developed. In our view, these two detailed analyses are the keys to success. A detailed front end analysis is the key to

achieving any successful training program, but particularly total training system acquisitions. A major lesson learned is that front-end analysis data should be delivered on magnetic disks and that it be IBM compatible.

STATEMENT OF WORK

Using the front-end analysis as a baseline, we developed the statement of work. A training system statement of work should be designed to allow contractor flexibility in his approach while at the same time defining those constraints within which he must operate. It should include system baseline and training requirements analysis as part of the package. It should also detail site activation requirements and the requirement to develop and maintain a recompetition package, and require concurrency management of the hardware and software within the system. These basic factors are very important in providing the basic structure from which a contractor should build his proposal.

SYSTEM SPECIFICATION

The most important factors in developing our system specification, from the user viewpoint, were the fidelity requirements of the computer based training and the system response time as it related to actual aircraft procedures and systems operations. Additionally, from the supporting command's viewpoint, phasing of modifications for the weapon system trainer was the major issue. Timing of modification installation was significant, particularly as it related to ongoing aircraft modifications.

INSTRUCTIONS TO OFFERORS

The development of instructions to offerors was based on the statement of work and system specification in an almost item by item fashion. It is very important that the instruction to offerors follow the statement of work and system specification in order to facilitate tracking in the evaluation process.

STANDARDS

Very few hard standards were established for the C-130 ATS acquisition. Basically our approach was that the contractor's proposal had to be reasonable and adequate in relation to the job that was required. What this means in a broad sense is that specific standards are defined by the contractor's proposal itself and become part of the contract; therefore, binding. This understanding is particularly important when changes are made to the original proposal without explanation or justification during source selection.

ROLES AND RESPONSIBILITIES

In establishing evaluator roles and responsibilities it is vitally important to assign and correctly define the primary areas of expertise and responsibility. In the C-130 ATS, the procuring agency was primarily responsible for evaluating engineering aspects of the program. They were also responsible for contract management, for the contract itself, and for the cost portion of the proposal.

The using command should always be given responsibility for the operational arena. This should include course development, continuation training, training evaluation, and the management portion of the training system itself including operations and maintenance over the life cycle of the program.

The supporting command should be given responsibility for the logistics portion of the program. This includes depot functions, modifications (including design engineering), recompetition package requirements, and the costs associated with these elements. Additionally, those areas of contract management associated with tasks performed after program responsibility transfers from the acquisition agency should be the responsibility of the supporting command.

Diffusion of these responsibilities can lead to much confusion and turbulence in acquiring a functional system if not given careful consideration.

EVALUATOR SELECTION

In the evaluation of aircrew training systems the selection of evaluators is the most important area to be considered. Traditionally, in a hardware oriented environment the procurement agencies were the experts in all areas of the acquisition. These acquisitions were based primarily on MILSPECS and detailed engineering requirements.

If we are to achieve the same excellence in the training systems arena, however, acquisition policies in the selection and placement of evaluators must change in order to select experts from operational command and logistic support arenas. This is absolutely necessary in a total training system environment for two reasons.

First, each command or service has its own specific regulations and basic concepts in it's approach to training. Basic concepts and requirements will not be changed or abrogated in a contract environment. Overall, operational experts for each command are found within the command itself. This is particularly true when you are buying a training system for a weapon system that is already in being and has been for some time. In the case of contractor logistic support or management of modifications to hardware, the logistics command experts are the people who should select the evaluators and be given primary responsibilities for that area of the evaluation.

Second, the assignment of the evaluation team from each organization needs to be accomplished during the RFP preparation phase. These personnel should be permanent until the source selection phase is completed. This allows for continuity in the process and reduces program risks and time wasted for training of new personnel. This is especially important given the philosophy of an ATS program where MILSPECs are not necessarily the governing procedures.

EVALUATION DEFINITIONS

When evaluating proposals, basic definitions become very important in trying to establish ratings. Some definitions which are hardware oriented are very difficult to interpret when applied to an ISD environment. Black and white definitions for weaknesses and risks when applied against a reasonable or adequate standard can become very grey and are subject to much interpretation when evaluating a training system oriented proposal. A totally contracted training system approach using best commercial practices inherently leads towards more subjective standards. The bottom line is that in acquisition of contracted training systems, subjective evaluation is a fact. This is why it is so important to insure that your best people, who

are the best qualified experts you can find, are chosen and given the responsibilities for making evaluation decisions for their area of expertise.

CONTRACTOR NEGOTIATIONS

Face to face negotiations are important in the accomplishment of any acquisition. In the past, information provided to the contractors was very controlled, standardized, and an important part of hardware specification and acquisition. Strict adherence to this principle, in our view, is no longer feasible. Given RFP flexibility, each contractor is allowed to design and propose his own unique approach to training. This can lead to significant differences in contractor proposals which make standardized negotiations with each contractor very difficult, if not impossible.

This is not to say that we propose a total freeflow of information. We must still protect proprietary information and conduct information exchange within legal constraints. However, providing information to a contractor just because you discussed it with another contractor can be very confusing. In many cases it can create problems where there were none to begin with. Procurement policies must change on this issue if negotiations are to have any significant function at all.

OPERATIONAL ISSUES

There were five operational lessons learned in the C-130 ATS acquisition. The first was how flying hours should be life cycle costed. Flying hours are subject to verification in the summative evaluation phase of the program. Given this fact and that they may be changed during this period and that the baseline for the flying hour program will not be signed until the program reaches training system readiness review. It is our opinion, that future contractor cost evaluations should be based solely upon the proposed ground training program. The statement of work should require an optimized flying hour program based on the contractor's proposed ground training program and evaluated solely on technical content. This is a feasible approach as the cost of flying hours is a command operational & maintenance (O&M) requirement and the final decision on utilization of these hours falls within the using command's authority and not the contractor or procuring agency.

A second operational lesson learned was the inclusion of a requirement for timely courseware updates at no charge. This will allow changes to tech orders, flight manuals, etc. to be accomplished without having to go through a costing loop and/or a contract amendment every time a change comes along. Our approach was to require courseware and changes at a fixed price except for changes to mission or tactics. Operations and Management money should be maintained within the command budget structure to allow for these eventualities.

A third important lesson learned in the operational arena was the design of student throughput cost windows. These allow for surges in the student population in the formal school without incurring changes to the cost per student. There must be allowances; however, for costs incurred in acquiring additional instructors or equipment, given surges larger than those specified in the RFP.

The fourth lesson learned is to be absolutely clear about the level of detail to which you want your task and objectives developed. This is important in defining the courseware content and effectively accomplishing the ISD feedback loop. This detail ultimately will result in satisfying and tracking the training requirements of the guaranteed student.

The final operational lesson learned was to ensure that all the operational training volumes of the proposal were put on contract. This will ensure that risks are minimized during development and implementation given that the contractors proposal specifies the components and detailed requirements for the entire system.

LOGISTICS ISSUES

Because the C-130 program is a fully operational system with "Blue Suit" instructional and maintenance functions in place, it provides a unique challenge for a smooth transition from organic Air Force operation to complete contractor takeover. Facilities and other resources will have to be shared for a period of time until the transition is complete. Hence, flexibility was built into the RFP by providing a 6 month period for an orderly, efficient changeover. The drawback to this approach became evident with the delivery of existing spares and support equipment. With two separate organizations maintaining and supporting the same devices, spares and support equipment, must be shared by both parties. This can become unmanageable. Additionally, without specific guidelines on equipment transition, the possibility exists to divide the responsibility for devices housed within the same facility or even split the responsibility for equipment shift by shift. This creates serious problems for custodianship, security, and overall responsibility for workmanship.

With the formulation of the recompetition support package, a major lesson learned lies in the definition of what it should contain. This, of course, differs based upon each contractor's maintenance approach. The absence of specifics in the C-130 ATS program, resulted in as many approaches as proposals submitted. The idea of what is contained in government furnished equipment versus what the contractor should provide as a function of operations and maintenance was not standardized. This led to a great deal of confusion and an even broader spectrum of proposed approaches and resultant cost impacts.

Also, with the magnitude of government furnished equipment (GFE) being offered by the Air Force, the issue of serviceable versus reparable assets and the availability of bench stock items were important factors. Most GFE is peculiar to the C-130 system and therefore required for system support. On the other hand, bench stock items are already located at the main operating bases and are common to any like maintenance function; they are therefore listed in the RFP as residual assets. A breakout of this information should be required to provide guidelines to the competitors so that some standardization of the contents of the recompetition package will result.

With the transition to a contractor program, security implications and requirements must be analyzed and defined very early in the RFP preparation process. This includes equipment, facilities and instructional media, and ranges from a simple task like visitor control to complex tasks concerning classified data responsibilities. When

these requirements are not defined up front and subsequently appended after contract award, program objectives may be adversely impacted while necessary clearances are obtained. This can be a costly process if the contractor has already hired and located personnel for which clearances cannot be obtained in a timely manner.

In attempting to allow total flexibility of maintenance approaches within the C-130 ATS the depot repair function and operation of the Training System Support Center (TSSC) were not specified in detail. This created some unique problems for the Air Force and subsequently for the competitors since the C-130 ATS currently contains two TSSCs, one for the weapon system trainer (WST) and one for the visual system (VS). To add to this confusion, the Air Force VS TSSC is still being utilized by the original equipment manufacturer for completion of acquisition tasks.

The location of the depot repair function and its supply network were critical to this program due to worldwide locations of the Main Operating Bases (MOB) and the resultant restrictions placed upon goods being shipped into those countries. With the emphasis placed upon the "guaranteed student" as the product of the program, the equipment associated with continuation training could potentially suffer if the formal school requirements are overemphasized. Therefore, some basic government guidelines concerning depot and MOB supply functions could have enhanced the proposed approaches.

MANAGEMENT ISSUES

There were two lessons learned in the management arena. One was the proposed approaches for configuration management, including the required interfaces, and the other was the interaction of different organizations working together as a cohesive team.

It has been the goal of the Air Force to allow for recompetition on all levels of acquisition. Because of this, the concept of configuration management has taken on new meaning, especially in the realm of using best commercial practices. The specifics of how to achieve configuration management utilizing best commercial practices is difficult to understand for those of us that have very defined concepts from a MIL-STD point of view.

Additionally, since the C-130 ATS will continue through FY99, the challenge of concurrency management enters into the picture. This requirement means interface agreements with a wide range of organizations must be reached. These range from the Air Force depot (WR-ALC) to any number of contractors/sub-contractors. Compounding this challenge are the time requirements imposed on the contractor for the completion of training related changes while the authority for the changes resides with the Air Force. The interface requirements and procedures established between the Air Force and the contractor become absolutely critical at this point.

Because the C-130 program is a tri-command effort (AFSC, AFLC, and MAC), working relationships are a key factor in achieving the ATS goal. Due to the longevity of the contract itself, each involved organization must be sensitive to the requirements of the other. Adequate review time and coordination must be allowed to insure that each agency's expertise is fully used. This will result in an increase in the effectiveness of the RFP while decreasing the effort and time required to complete the source selection due to changes or errors discovered after release of the RFP.

SUMMARY

The intent of this paper was to point out potential pitfalls and how they can be avoided with early planning, close coordination, and an understanding of vital concerns by using and supporting command. These concerns can aid in the future acquisition and procurement of quality training programs within the government; if acted upon.

We understand that government source selections must be conducted within current regulation guidelines. However, it may be time to revise these regulations to incorporate "training systems" acquisitions and new advanced approaches in training philosophy. It may also be time to give using and supporting commands an equal say in the acquisition of total training systems.

Our intention is to pass along the benefits of many long hours and hard work to those of you who may soon be facing these same challenges. We feel that these lessons learned will help significantly in achieving the critical advantage of a total training systems approach while maintaining the lowest possible risk to the using commands and ultimately their combat capability.

ABOUT THE AUTHORS

Lt Col Dukes has just assumed duties as the Chief of the HQ MAC OL which is primarily responsible for new courseware development, quality assurance and development of the training management system for the C-130 ATS. He is a command pilot with over 3,000 hours of flying time in C-130 aircraft. His background includes time in tactical airlift, special operations, avionics maintenance, formal school instruction/ standardization, and 4 years in the Directorate of Aircrew Training and Resource Management at Headquarters Military Airlift Command.

Ms Jo Voeller is the AFLC Program Manager for the C-130 training system. She is primarily responsible for the contractor logistics support program and concurrency requirements of the training equipment. She has been working as a training device system program manager since the formulation of that division at Ogden Air Logistics Center in 1979 and has been directly responsible for HQ MAC programs involving the C-5 and C-141 training devices. She is currently responsible for HQ MAC programs involving the C-130, C-135, CH-3, HH-53, and MH-53 training devices. In addition, she was responsible for the implementation of the program for conversion of the 341XX AFSC career field to contract maintenance for F-4, F-111, F-15, F-16, and A-10 training devices.

THE C-5 AIRCREW TRAINING SYSTEM (ATS):
A USER PERSPECTIVE OF THE ADVANTAGES AND PROBLEMS

Major Jonathan D. Baethge
Chief, C-5 Aircrew Training System
Directorate of Aircrew Training and Resource Management
Headquarters Military Airlift Command
Scott Air Force Base, Illinois

ABSTRACT

Military Airlift Command's (MAC) first aircrew training system (ATS) becomes fully operational this year. As we implement the C-5 ATS, we are keenly aware of the significant advantages of contractor provided training. Best commercial practices provided MAC new FAA Phase II Weapon System Trainers ready for training 23 months after contract award and an integrated ATS with formal and continuation training (24 courses, from initial qualification through flight examiner) in less than three years. Both acquisition and life cycle operation are at a substantial cost savings over in-house methods. As we begin training students, we can better define potential problem areas and user concerns. Crew members and leadership alike need to understand the training system and the key role they play in life cycle feedback. Quality assurance evaluation plans must be drafted and coordinated early to assure Air Force and contractor plans complement each other. Air Force procedures to approve, fund, and proceed with modifications must be streamlined to facilitate concurrency of the training system.

INTRODUCTION

In the summer of 1982, both the Air Force Scientific Advisory Board and the MAC Aircrew Training Task Force took a thorough look at the way MAC crews were trained. Their recommendations included immediate replacement of the existing C-5 simulators with new equipment, at least comparable to FAA Phase II standards. In addition, they suggested we take advantage of state-of-the-art training methods, including self paced strategies and computer aided instruction. At approximately the same time, the Air Force announced the acquisition of 50 additional C-5 aircraft. The additional crews associated with the new aircraft exceeded existing training capability. However, the funding associated with the aircraft buy provided an opportunity to update all C-5 training capacity and technology. Under the C-5 ATS, MAC sought to acquire contracted aircrew training by specifying the level of training and the desired crewmember qualification rather than training hardware. As the ATS went out for bids certain constraints were placed on the prospective contractors: (1) The courses had to train crew members to operate both C-5A and C-5B aircraft IAW MAC standards. Preliminary task and standard documents were provided to the bidders as guidelines; (2) In designing the training program, one of the stated goals was to provide qualified crews while minimizing the need for actual aircraft use; (3) The delivered training system was to contain all necessary instructional material as well as the logistic support and design data needed for 15 years of operation; (4) Training was to be accomplished at Altus AFB OK, Travis AFB CA, and Dover AFB DE; (5) To reduce development costs, the existing C-5 simulators were offered to the contractor as government furnished property; (6) All work had to be accomplished and the system ready for full operation by the end of FY88; (7) Total maintenance is the contractor's responsibility with minimum government intervention. The Air Force evaluation process was completed, and on November 6, 1984, United Airlines Services Corp. was awarded the C-5 ATS contract.

DESCRIPTION OF THE TRAINING SYSTEM

The C-5 ATS is a system of personnel, hardware, software, and courseware that generates qualified C-5 aircrew members and maintenance engine run personnel. The system currently consists of 24 distinct courses for the MAC formal school, in-unit upgrade, maintenance engine run qualification, and annual simulator proficiency training. It includes six weapon system trainers (WST), four cockpit procedure trainers (CPT), two cargo door and cargo loading part task trainers (PTT), two special function trainers (SFT), a management information system (MIS), a computer aided instruction (CAI) system, a software support center, and a courseware support center. This integrated system will ground train the full range of tasks for C-5 pilots (except air refueling part task hands-on training), flight engineers, loadmasters, and maintenance engine run technicians. The end products are guaranteed qualified personnel. The contractor is responsible for the operation, maintenance, logistics support, and configuration management of the ATS.

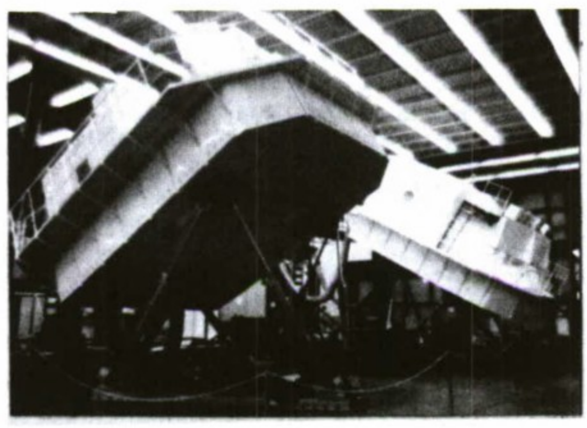


Figure 1 Two of six C-5B WSTs

CRITICAL ADVANTAGES

We are keenly aware of critical advantages of a contractor provided aircrew training system. Using best commercial practices, the C-5 ATS contractor was able to begin installation of the first of six C-5B WSTs at an Air Force installation just 20½ months after contract award. Twenty-three months after contract award, the WST had undergone Phase II certification by the FAA and was ready for training. All six were delivered in 29 months, at

major acquisition cost and time savings to the government. These new simulators provide MAC capabilities equal to or better than those in many commercial airlines.

In three years the contractor was able to provide MAC an integrated ATS with formal initial qualification, upgrade, and simulator continuation training. Key features include trainee proficiency advancement, pretests to determine where trainees enter the core curriculum and to identify the need for remedial training prior to entering a formal course of instruction. The contractor provides the dedicated instructional development resources and expertise necessary to make course changes as well as the engineering expertise to make equipment modifications, thus freeing up Air Force resources for warskill related tasks.

There are many additional advantages to contracted training systems. They are, in fact, too numerous to cover in the scope of this paper, and many have been adequately covered in previous I/ITEC-I/ITSC proceedings. The remainder of this paper will address problem areas and lessons learned, so that others may also achieve the critical advantage offered by training systems, but with greater ease, less risk, and without retracting our footsteps where we went astray.

LESSONS LEARNED

Consider all users. When stating the training requirements, remember all users for whom training has been provided in the past. The primary users are obvious, but have you considered the other DOD agencies who send only a few trainees annually? Obviously it's not cost-effective to build a training system that will do everything for everybody, but allow the contractor the flexibility to meet the training/scheduling requirements of unique agencies if he can reasonably do so. Involve all users early in the planning phase so that the statement of work is complete and truly states the user's training requirements.

Initial training system evaluation. We frequently hear it said that what we want out of a training system is a trained student, and the true test is a user conducted evaluation of the student to ensure that the training system produced what we paid for. However, most of us would agree that a four hour evaluation at the end of eight weeks of training could only hope to look at a very small percentage of the total training objectives. The true test of training is, rather, whether the training system graduate can perform at the intended level of competency while on the job. On-the-job situations can be quite different and conditions far more complex than those found in the controlled environment of most end-of-course evaluations. Unfortunately, on-the-job performance evaluations may not be possible until months after a contracted training system is brought on line, and valid trend data may not be available for much longer.

By doing a thorough and careful evaluation of the instruction within the training program, the probability of obtaining the required on-the-job performance can be closely predicted, provided course content is evaluated in terms of job performance tasks and standards.

In contracted training programs like the C-5 ATS, the contractor is responsible for providing an evaluation of the training system. In order to minimize risk to the user, delivery of the quality control and summative evaluation plan should be requested well in advance of training delivery. The key to a successful training system lies first in top quality job-related instruction, but is followed closely by a thorough summative evaluation process which provides timely feedback and self-correcting features throughout the life cycle of the system.

Courseware readiness and implementation. "Turnkey" startup of student training under a contracted training system allows the contractor to make a clean break from the old methods of training to the new contractor-provided methods and media. On a specified day, the previous training equipment and curriculum are shut down and the contractor training system is turned on. There can be disadvantages to that approach in that it tends to ignore the lessons learned in years of operation of the previous training system. Takeover of existing courseware and training, with a requirement to develop state-of-the-art instruction may be an alternative. The contractor can develop new courseware and procure new training equipment while operating the existing training program. Lessons learned about student profiles, academic weaknesses, subject difficulty, job tasks, etc., while conducting the existing training could be incorporated into the training system during development.

Concurrent courseware. One goal of any training system should be to provide training courseware that is concurrent with procedure, equipment, and task changes. In other words, if major procedural changes will occur next Monday, the training system should begin training the new procedures next Monday, not six months from now. Courseware changes require ample lead time, therefore training system managers need to work out advance notice agreements with all agencies that have the potential to impact training. Likewise, training managers need contractual tools to proceed quickly with changes.

Besides lead time, major courseware changes require "dollars," and "dollars" generally require bureaucratic approval. The best of courseware maintenance systems cannot keep training current if the user cannot cut through the red tape to approve and fund changes. In general, user organizations should commit to funding and approving training changes when they commit to making a procedural, equipment, task or any other type change that impacts the training system. At that point, the courseware change should be approved, with only the price to be negotiated.

Courseware configuration documentation. Configuration documentation and control are absolutely essential in any large training system. In the C-5 ATS, with nearly 5,000 crew performance objectives (CPOs), it's essential to cross reference every CPO to the courseware. Likewise, crew tasks must be cross referenced to the lessons where they are taught, and a lesson index must list all tasks taught within all lessons. With a change in a basic procedure, dozens of lessons in computer delivered training, flight simulators, and aircraft flights might have to be revised. Without a comprehensive cross reference data retrieval system, affected lessons, student handouts and visual aids can't be easily found. Without the courseware cross reference system in the C-5 ATS, contractor and customer quality control would be a nightmare.

Establish user acceptance. Management should strive to establish user acceptance (down to the lowest level) of a training system in advance of

actual implementation, in order to help achieve the critical advantage. This can be done through newsletters, roadshows, official correspondence, etc. But, perhaps one of the most effective methods to "spread the word" is to publish a user's guide to the training system, complete with background, operational requirements, and a description of user participation in modifications to the system. Above all, tell it like it is and listen to the feedback.

Design training facilities alike. If a training system is required to deliver instruction at more than one site, those sites having identical training equipment should be designed by the user with as much facility standardization as possible. If a delay occurs in facility construction at one site, equipment originally scheduled for installation there can be diverted to another facility. If the facilities were designed identically, the diversion can take place without extensive engineering studies, modification, and recabling, etc.

Approach with an open mind. Approach a contractor conducted training task analysis with an open mind. Understand that some of the tasks which we historically trained weren't based on any sound training requirement. They were taught "because we've always done it that way." A thorough task analysis resulting in a Master Task List/Evaluation Standards Document (MTL/ESD) can provide much better definition of the actual training requirement. However, be aware that tasks critical to the job may be omitted through oversight or through the external decision-making process. When that occurs in the aircrew training arena, the user must remember that some training lessons were learned the hard way, with fatal accidents and destroyed airframes. Critical tasks and standards and safety cannot be compromised. Therefore, we, the users have to recognize our own experience in training and not sell ourselves short, just because we've hired a contractor to design a training system. While we stand to learn a great deal from commercial training practices, we the user have also learned a lot over the years. When we have a better idea, we need to stand up and be heard.

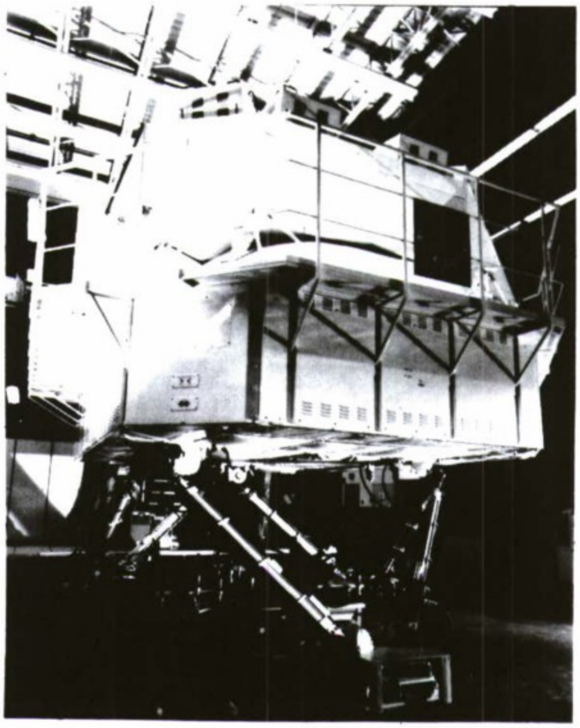
Provide specification for Computer Based Instruction response. Computer Based Instruction (CBI) provides many advantages to classroom, lecture-oriented instruction, however many of these advantages can be negated if the computer terminal screen response time is too slow. In the interest of not restraining the contractor, the C-5 ATS system specification did not state a limiting response time to advance from one screen of computer text or graphics to the next. Consequently, the contractor is currently attempting to achieve (through hardware and software modifications) a response time which is acceptable to the students. While CBI delivery systems are available today with very fast response times, we are still driving to achieve a reasonable goal after nine months of operation. There are tradeoffs to be considered when selecting a CBI system; system response time is probably one we should have specified.

SUMMARY

The purpose of this paper was to highlight some of the critical advantages of contractor provided training systems, but also to point out C-5 lessons learned so others who follow may do so with less risk. Many problems can be avoided through thorough planning, open communications,and a willingness to try something new. In spite of some growing pains, contracted training systems have tremendous advantages to the military in terms of dollar and time savings.

ABOUT THE AUTHOR

Major John Baethge is the C-5 ATS Program Manager at Headquarters, Military Airlift Command, Directorate of Aircrew Training and Resource Management. He is a senior pilot with over 3500 hours in C-5, T-39, and C-131 aircraft in Military Airlift Command and Air Force Systems Command. Major Baethge holds a Bachelor of Science Degree in Electrical Engineering from the University of Texas at Austin and a Masters Degree in Management from Webster University.



EMBEDDED TRAINING
THE ARMY'S DILEMMA

Captain Alvie Johnson
U.S. Army Combined Arms Training Activity
Combined Arms Training Support Directorate
Fort Leavenworth, Kansas 66027-7000

ABSTRACT

The requirement to train in peace and war continues to exist. Soldiers and units that deploy to combat with equipment which contains an embedded training capability will possess the tools necessary to sustain proficiency in conjunction with combat operations. Further, peacetime constraints on individual and collective training caused by time, space and resource shortfalls are expected to continue. Therefore, the Army must pursue the operational systems trainers in the development and product improvement programs for operational systems. The acquisition of hardware systems that have an embedded training capability will offer the Army an opportunity to train at all echelons while in garrison or in the field.

This paper will examine how senior Army leadership/guidance has identified the need for embedded trainers in operational systems and will provide the insight into why this need has not been fulfilled. Embedded training will then be defined from a military perspective, and will identify the four essential categories of embedded training that must be considered to ensure operational readiness can be maintained. The Army's materiel acquisition process, and Training Developers role will be examined.

Finally, this paper will conclude with a listing of actions which must occur during the acquisition process. The ultimate goal of this paper is to ensure embedded training becomes the first alternative for training by the materiel, combat and training developer.

INTRODUCTION

The first exploration of the use of embedded training occurred in the 1950's with the development of the Semi Automatic Ground Environment (SAGE) System for the US Air Force. The SAGE's successful employment of embedded training technology was one lesson which was not learned. Therefore, a potential for increased training was lost, and some thirty years later the need for embedded training was rediscovered.

During recent years the use of appended or stand alone training devices has been the primary method for training on materiel systems. This was due partly to convenience. Identification of the training device need could be accomplished early-on (many times wasn't) during the formulation of the Operational and Organizational (O&O) plan. The actual training device requirement was not required until much later. Too often,the hardware system was on its way to the production phase of the Life Cycle System Management Model (LCSMM) when the training device need statement was developed for that system. This caused the training of that system to lag behind the fielding.

The publication of The Army Plan in December 1985 established the initial need for the Army to pursue the use of advanced technologies as a means of altering and improving training. This need was further substantiated by the Army Science Board Report released in 1985 which stated: Embedded Training/Testing opportunities and training simplification are largely unexploited in the materiel development process.

The Army Science Board provided the following recommendations:

- That Training and Doctrine Command (TRADOC) and the Army Materiel Command (AMC) require system specifications to include training simplification from concept through test, evaluation and Pre-Planned Product Improvement (P3I), and provide incentives to program managers and contractors.

- That Project Manager Training Devices/Army Research Institute accelerate study examining present and planned Embedded Training in the Army.

- That TRADOC and AMC utilize the LHX Embedded Training System as test bed for development of implementation policies, concepts, and its application to other systems.

The Army Science Board's recommendation for the use of Embedded Training, coupled with the identification of the need for Embedded Training in numerous Army publications, prompted many senior personnel to relook the need for Embedded Training in emerging materiel systems.

Before the full potential of embedded training could be realized it had to be defined. Lacking a clear definition of Embedded Training, many trainers and combat developers were unable to effectively articulate their desires/requirements to the materiel developers. More commonly, the need for embedding training was so poorly conveyed that we got exactly what we asked for, which wasn't what we wanted or needed. Therefore, it was evident to the Senior Army leadership that we needed to define embedded training to clear up the confusion, and also energize the Army to consider the use of Embedded Training within the current Materiel Acquisition Process.

Embedded training is defined as training that is provided by capabilities designed to be built into or added into operational systems to enhance and maintain the skill proficiency necessary to operate and maintain that equipment end item.

Embedded Training:

a. Will not adversely impact the operational requirements/ capabilities of the system and should be identified early enough to be incorporated into initial prototype designs.

b. May train individual tasks through force-level collective tasks as required. Embedded trainers encompass four training categories:

(1) Category A - Individual/Operator Training Objective: To attain and sustain individual, maintenance and system orientation skills.

(2) Category B - Crew Training Objective: To sustain combat ready crews/teams. This category builds on skills acquired from Category A.

(3) Category C - Functional Training Objective: To train or sustain commanders, staffs, and crews/teams within each functional area to be utilized in their operational role.

(4) Category D - Force Level (Combined Arms Command and Battle Staff) Training Objective: To train or sustain combat ready commanders and battle staffs utilizing the operational system in its combat operational role.

System Development

The Concept Based Requirement System (CBRS) is a systematic and flexible approach to determining future Army needs and resolving deficiencies in current battlefield capabilities. Based upon analyses of Army and Threat capabilities, historical lessons, Army mission and doctrine, and emerging technologies, the Army identifies and prioritizes operational deficiencies. The CBRS identifies solutions in four areas: Doctrine, Training, Force Structure, and Materiel. A materiel solution is generally the most expensive and requires the longest lead time, and is the last choice for resolution of deficiencies.

Once the Army determines that a materiel solution is required, a concept is developed which describes equipment performance parameters. Simultaneously, a training concept is established based upon comparison with existing systems and any known or anticipated training constraints. This information is gathered in the O&O Plan which, once approved, allows a program to enter into the Life Cycle System Management Model (LCSMM).

The LCSMM is not a rigid process. It provides an integrated developmental framework within which a program progresses through concept, fielding and support. System managers and developers may tailor a developmental program depending upon risk, cost, and priority. However, the general framework and intent of the LCSMM will still be valid. Upon entry into the LCSMM, the combat developer, training developer, logistician, and human factors engineer will initiate a set of analyses to determine the most cost effective approach to designing a new system. The actual system requirements are defined in the Required Operational Capability (ROC). The ROC describes the minimum essential operational, support and cost requirements for a new system. These requirements are the basis for the developmental contracts established to engineer and procure a new system.

Integration of Embedded Training

The Concept Based Requirement System (CBRS) and the Life Cycle System Management Model are two dynamic methods (processes) which, if performed correctly by the participants, will produce materiel systems which are logistically supportable, trainable and operationally capable. These processes require honest and careful consideration of the use of Embedded Training to be pursued during concept formulation and through the process by all the players (training developers, combat developers, materiel developers). The first consideration for embedded training must occur during the CBRS, specifically during concept development when we are listing the potential materiel solutions. Normally, this function is performed by the combat developer. The Training Developer must become an active participant at this point to get the earliest consideration for embedded training. The consideration of "how we fight" with a system must go hand-in-hand with "how we train" that system. Thinking through the training needs, early on, enhances the chances of developing a complete training package for a system.

Prior to leaving the CBRS process, the manpower and personnel integration (MANPRINT) process begins. MANPRINT is an umbrella program integrating human factors engineering, manpower, personnel, training, health hazard and system safety into the materiel acquisition process. It influences materiel systems design so that systems can be effectively and safely operated and maintained with the manpower structure, personnel skill, and training constraints of the Army. To manage the MANPRINT issues during the materiel acquisition process a MANPRINT Joint Working Group (MJWG) is established.The purpose of the MJWG is to manage MANPRINT issues during the materiel acquisition process. It consists of members from the proponent Directorates of Combat Developments, Training and Doctrine, Evaluation and Standardization, Safety Office, Proponency Office, Integration Center, Supporting Schools and AMC Mission Area Manager. The MJWG is responsible for developing a System MANPRINT Management Plan (SMMP) for all development,

nondevelopment and product improved systems. This is a living document that will be updated as needed throughout the materiel acquisition process.

The SMMP serves as a management tool and audit trail which identifies tasks, analyses, trade-offs and decisions that address MANPRINT issues. During the formulation of the SMMP the training developer must identify the training concept, training constraints, and training issues and criteria. This is the point at which the need for Embedded Training must be integrated into the training concept. The SMMP feeds all materiel acquisition documents throughout the process, therefore all training needs must be identified by the training developer and managed by the MJWG throughout the Acquisition Process to ensure the needs are addressed.

Once the concept formulation is completed and before a system can enter the LCSMM, an O&O Plan must be developed. The O&O Plan tells what deficiencies this materiel system will eliminate,who, how and where we plan to use this system. The combat developer is the key person who coordinates with the materiel developer, training developer, transportability agent, logistician, MANPRINT planner, tester, evaluator and interested Major Commands. This is the point where the need for Embedded Training must be identified and written in the plan as a constraint. The SMMP will serve as feeder data for the Combat Developer during the development of the Operational and Organization Plan. Therefore, if the need for embedded training has been identified as a constraint during development of SMMP and identified as part of the training concept, this information will then be utilized in the development of the constraints paragraph. This information will also be utilized in the various analyses conducted to determine the most cost effective approach to designing the training for this system.

All documents prior to the development of systems requirements are merely plans which invariably change. When the decision is made on what the actual system requirements are, the Required Operational Capability (ROC) must be developed. The ROC is the Army's definitive statement describing the materiel solution to a mission area deficiency defined through the CBRS. It states the minimum essential operational, MANPRINT, training, logistical, technical, and cost information to initiate engineering and or operational systems development or acquisition of the materiel solution. The key players involved in the development of the ROC are the combat developer, training developer, Rationalization Standardization and Interoperability manager, logistician, MANPRINT planner, tester, evaluator and interested MACOM.

The ROC is a formal requirement. If embedded training is identified as a part of the systems training strategy, then it must be identified in the operational characteristics and training assessment paragraphs. By identifying the requirements for embedded training as an essential characteristic of the system causes embedded training to be developed as part of the operational system. The lead document which provides the input for the Combat Developer is the SMMP. The Training Developer is still responsible for ensuring the requirement for embedded training is identified in the ROC. The ROC clearly is the last time the Army has an opportunity to influence the design of a system. If the training developer fails to identify the need, any attempt past this point in the acquisition process to provide embedded training is probably cost prohibitive.

Requirements identified in the ROC must be written in the Request for Proposal/Statement of Work (RFP/SOW). RFP/SOW provides total system requirements including soldier performance and force assessment. This document is developed by the Materiel Developer with input from the Combat Developer and Training Developer. The role of the Combat Developer and Training Developer is to ensure that the requirements,constraints and assessments identified in the ROC, O&O plan,and SMMP are articulated well in the RFP/SOW.

CONCLUSION

As we look toward the future of embedded training we can not overlook the Armored Family of Vehicle Program. It is a program which has developed a fully integrated training program that utilizes a mixture of devices and simulations making every attempt to embed training wherever possible. This training addresses individual, collective and sustainment training needs in both the unit and institution. The potential for embedded training being a positive force in this program is because the forethought was there in the initial identification of the training requirement for the system.

In conclusion, training has to be considered and evaluated throughout the life of a system. Any requirements for embedded training should be identified in the SMMP, specified in requirements documents, and validated during supporting analyses. Developmental contracts and prototypes, and production contracts should require,at minimum, embedding of training capabilities. The concept and developmental work must be done early in the LCSMM in order to ensure inclusion in the final design and production. The requirement to train in peace and war continues to exist. Therefore, units that deploy to combat with an embedded training capability will possess the tools necessary to sustain proficiency in conjunction with combat operations.

REFERENCES

- How to Select and Develop Embedded Training: Overview of Interim Guidelines, Procedures and Supporting Documentation, Dorothy L. Finley and Irving N. Alderman, U. S. Army Research Institute, Donald S.

Peckham, Project Manager for Training Devices, H. C. Strasel, HiTech Systems, Inc., March 1987.

- AMC-TRADOC Materiel Acquisition Handbook 70-2, HQ, U. S. Army Materiel Command and HQ, U. S. Army Training and Doctrine Command, 26 March 1987.

- Materiel Objective and Requirements, Army Regulation 71-9, HQ, Department of the Army, 20 February 1987.

- Manpower and Personnel Integration (MANPRINT), Army Regulation 602-2, HQ, Department of the Army, 21 July 1986.

ABOUT THE AUTHOR

Captain Alvie Johnson is the Project Manager for Signal Systems at the Combined Arms Training Activity, Combined Arms Center, Fort Leavenworth, Kansas. Also serves as the Training Developer for the Maneuver Control Systems. His previous assignments include assignments to Fort Bragg, North Carolina and Fort Polk, Louisiana where he served as Platoon Leader, Company Commander and Battalion S-3. Is a graduate of the Combined Arms and Services Staff School (CAS3), Signal Officer Advanced Course (SOAC), Signal Officer Basic Course (SOBC). Captain Johnson is a distinguished military graduate from Howard University with a Bachelor of Arts in Broadcast Production with emphasis in television.

FUTURE TRAINING WITH
THE ARMORED FAMILY OF VEHICLES

MG(Ret) Robert J. Sunell, USA, and
MAJ(P) Thomas R. Rozman, USA
Armored Family of Vehicles Task Force
Fort Eustis, Virginia 23604-5597

ABSTRACT

The Armored Family of Vehicles (AFV) is programed to replace the current fleet of armored vehicles beginning in the late 1990's. The AFV will consist of at least two chassis (medium and heavy) that will accommodate appropriate modules for combat mission requirements. Significant features of the AFV are commonality and advanced technology which will provide a once in a lifetime opportunity to develop, test, and validate a mature and coherent training subsystem simultaneously with the new equipment. This envisioned training subsystem offers a potential to maintain high training readiness at less cost than in the past. As systems become more complex and costly, this objective becomes more critical.

INTRODUCTION

Development of an AFV presents the ground component of our Armed Forces with a unique training opportunity. This opportunity is the potential to attain and sustain a high state of training (operational) readiness despite budgetary or local constraints. This can be achieved through a training subsystem designed and developed as an organic part of that family. A key element of this subsystem is simulation technology. Most intriguing is the promise this approach offers for more efficient use of increasingly scarce training resources.

WHAT IS AN AFV?

A Mounted Force. To set the stage, I will describe the AFV. In the past, the development of mounted armored forces has proceeded on a one at a time or "eaches" format. As the tank matured and other arms, such as infantry and artillery, developed defensive measures that compromised the tank's maneuverability and protection, it became apparent that other capabilities needed to be mounted and armored along with the tank. To achieve a balanced combined arms team, a staggered introduction of infantry carriers, reconnaissance vehicles, armored self-propelled artillery and command and control vehicles resulted. The perennial effort to keep this assortment of vehicles competitive with potential opponents caused armies to introduce improved systems. In most cases, new generation vehicles were unique systems sharing minimal commonality with other fielded armored vehicles.

Occasionally, an effort was made to build more than one system from a single chassis creating a rudimentary sort of vehicle family. Often, in such cases, the impetus was on availability of the chassis. There were, however, examples of an attempt at achieving the intuitive efficiencies of a range of vehicles constructed from the same chassis, i.e., the M113.

The result of this approach to armored vehicle development was a fleet that, at a given time, could possess up to six or more different armored chassis. Each new generation replacement was more costly than its predecessor. Arms (branches) that had not had major forward roles on the early armored battlefield increasingly found themselves in this area, i.e., air defense and military intelligence, and would attempt to execute their missions initially in soft skinned wheeled vehicles. As existing armored chassis became available, they would be adapted to support these missions. Not infrequently, the chassis would be an older generation vehicle.

The end result was a mounted force that evolved. It was not designed from the ground up using an integrated all arms concept. True, funding and necessity played a constraining role in this process. The most critical players were equipped first with the armored fleet of the day consisting of groups of unique vehicles.

This practice led to inconsistencies in performance between vehicles. Examples of the consequences of separate design were infantry carriers that could not keep up with tanks and differeing protection levels that precluded vehicles that needed to operate together from doing so. Other side effects were extensive spare parts stockage levels and varied and complex training needs.

Commonality and Synergy. The AFV, through a system of (see figure 1) simultaneously designed vehicles, will employ as much commonality from mission design to mission design as possible. The objective is to obtain maximum benefits from synergy. The resulting force is expected to be a fully integrated, highly survivable, combined arms organization, more lethal to the enemy, more cost efficiently sustainable by the Army, and able to accept improvements as a systematic, planned process.

ARMORED FAMILY OF VEHICLES

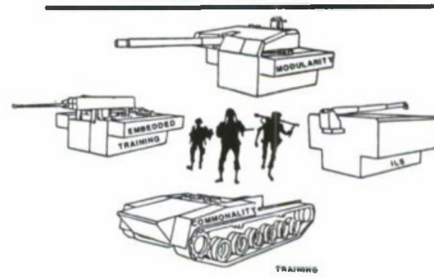


Figure 1

A key synergistic effect that commonality and simultaneous design and production of mission systems is expected to provide is reduced operating and sustainment costs. It is in this area that the training subsystem is expected to have its greatest impact.

Incorporation of Simulation Technology. An essential element that must be available if the Army is to obtain the advantages just discussed is simulation technology. Current and future developments in this field indicate the ability to replicate individual, crew, and collective operational tasks with a high degree of fidelity. There are excellent indications that the state of training technolgy, combined with advances in vetronics, very high

speed integrated circuits (VHSIC), and common module fire control can provide the ability to build this simulation capability directly into the operational equipment at minimal, if any, cost to operational function. A critical point to note is that if we are to capture this training capability, we must cause engineers to design it into the hardware at the outset. Failure to do so places the capability at risk in a budget prioritization process and, because it becomes a separate "add-on" or "stand-alone" subsystem, it can be "down" prioritized consistently below the funding line. Key to note is the lag time in fielding training systems. Currently, a training subsystem follows a major piece of equipment by 5 to 10 years.

AFV TRAINING CONCEPT

OPTEMPO and STRAC. A keystone--OPTEMPO is an Army acronym describing a process that sets mileage levels for each vehicle of the fleet for the training year. This mileage level then translates into supporting inputs of petroleum products and replacement parts. STRAC is a document that describes strategies for units to achieve weapon and gunnery proficiency. It then states the resource levels necessary to attain these proficiency objectives. It is in these three component areas that the Army expends a major portion of its annual operating budget for a given weapons system. This is as it should be because high training readiness is vital to a credible national security policy. Maintenance of such a state of training readiness demands individual soldier and crew equipment proficiency on a level that historically could only be attempted by actual maneuver or live fire exercises. Note: In the past, we have attempted to achieve high training readiness through full-scale maneuver and live fire training. We quickly learned that there are too many constraints and some training is too dangerous (emergency procedures, live fire, etc.). We learned that training devices that compliment live fire training better support training readiness. This OPTEMPO/STRAC supported training requirement today, and under any foreseeable system, remains critical to mounted force training readiness. However, as a practical concern, application of OPTEMPO/STRAC must be made more efficient and effective. It is entirely possible and feasible to enhance the proficiency of crews and units in their battle skills more precisely than is currently possible through OPTEMPO/STRAC supported exercises. This can be accomplished by using a new generation of mature simulation.

Device Supported Training. The AFV concept of training accepts the importance of maintaining a strong OPTEMPO/STRAC supported component. It also recognizes that the resource elements that support OPTEMPO/STRAC consume significant portions of the Army's annual budget. Additionally, future maintenance of current numerical levels of munitions to meet defined training strategies will require higher levels of funds. The reason is illustrated by the cost difference as we go to upgraded systems and munitions to defeat improved threat capabilities, i.e., a 105mm TP-T tank round of ammunition costs \$209 while a 120mm TP-T tank round of ammunition costs \$1,480. This is not a revelation. The Army has been moving to address this cost challenge by considering a training strategy that combines the most effective and economical use of ammunition, maneuver, and simulation.

The Conduct of Fire Trainer (COFT) is an example of what can be done. The basis of issue of one device to a battalion resulted in each tank of the battalion having its yearly maingun ammunition allocation reduced from 134 to 100 rounds. The COFT is a computer-simulation training device that places the vehicle commander and gunner in a fighting compartment virtually identical to the one in their vehicle. Through the use of digitized graphics and software scenarios, the crew is presented with dynamic terrain and target displays through their sights. The result is that they can execute all battle procedures necessary to engage and kill targets under a full range of conditions. Failure to do so properly (a miss) or using too much time results in engagement of their vehicle by the target. The system provides performance feedback and a record of performance that allows correction of deficiencies and charts progress.

The COFT replicates the gunnery experience impressively well. However, the device is a stand-alone and issued on the basis of one to a battalion. This means only one crew can exercise at a time. The driver and loader are not integrated into the simulation. Also, these devices do not net to allow collective formation-level simulations. While the COFT does fill a significant requirement, the concept must be expanded into 21st Century applications.

Technological Status. Current state of the art as demonstrated at Fort Knox where a stand-alone simulation extends this capability to the exercise of all crew members allows operations up to platoon level and maneuver against opposing forces on a 50km X 50km piece of terrain. This simulation is called SIMNET. Up to company level exercise capability is possible and planned. Considering this demonstrated capacity and noting the status of vehicular automated information architecture (Vetronics), VHSIC, and modular fire control technologies, it is reasonable to expect that the embedding of these capabilities into the vehicle is within reach. The layering of other computer capabilities on an existent architecture is an established accomplishment in the avionics arena. Avionics architecture has numerous examples of such layering, i.e., the sensing capability on the Apache system. Noting that Vetronics technology essentially derives from avionics, the layering of a simulation training system is an attainable objective.

Simulation's Role with AFV. Understanding that OPTEMPO costs will increase over time for constant levels of OPTEMPO, what can be done to sustain high training readiness if resources become constrained? It is essential that a simulation training capability be embedded into the vehicle that allows (see figure 2)

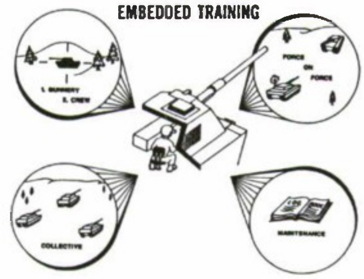


Figure 2

individual, crew (gunnery), collective (platoon, company), maintenance (log book), and force on force training. This capability developed to a high level of fidelity will, in many cases, exceed what

OPTEMPO/STRAC can provide. An example would be a greater density of replication of crew functions with feedback. Another benefit would be better opposing force capability supported by actual engagements and simulated kills that put crews out of action until appropriate measures are effected to replace or bring them back up. In terms of unit training, the concept would function as shown by figure 3.

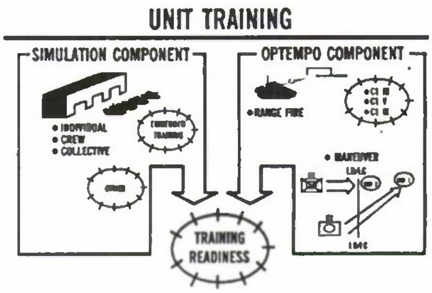


Figure 3

This embedded capability would be supplemented by a stand-alone simulation subsystem in the institution and Reserve Components that duplicates the embedded system (see figure 4). This subsystem offers the potential to reduce standing equipment requirements at the institution with attendant fuel, ammunition, and spare parts savings. It will also offer potential to develop reductions in the institutional overhead costs, i.e., operational equipment requirements and instructor staffs.

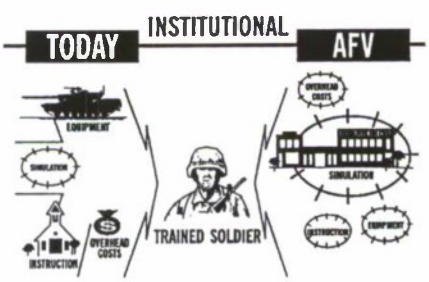


Figure 4

This simulation training subsystem will provide a virtually perfect built-in training capability. It will allow units to define an initial training objective, i.e., each crew achieves some level of first round hits, brief instruction periods by leaders, hands-on practice, execution (feedback), and definition of the next iteration's training objective. This is an excellent example of the performance-oriented training concept.

IMPACT OF AFV TRAINING CONCEPT

Efficiency. The realization of this concept, a conservative case, offers a potential savings of 24 percent on OPTEMPO/STRAC resource costs for a tank battalion. This is accomplished by substituting one simulation exercise for each type of OPTEMPO supported exercise currently defined by Army training guidance. The unit retains one or more iterations of each type of OPTEMPO/STRAC supported exercise. In most cases, more than one iteration remains.

Sustainment of High Readiness. The ready access of units to this organic simulation capability means that the unit can exercise crews with a high level of combat situation realism on a frequency never before possible. Unprogramed decrements in OPTEMPO/STRAC resourcing, though still having a severe impact on training readiness, will not totally cripple units in their efforts to maintain gunnery, platoon, and company tactical mounted proficiency.

CONCLUSION--TAKING ADVANTAGE OF THE OPPORTUNITY

Assured higher costs of all training related materials in any next generation combat systems, particularly the annually occurring fuel, repair parts, and training ammunition costs, represent a potential burden that from time to time may become simply unaffordable. Technology can provide, through simulation, an acceptable method of assuring a highly effective training subsystem that can take the initiative against spiraling costs. This is particularly true in the embedded format. This technology indicates an ability for units to sustain high levels of training readiness despite constraints. Such a system, walking hand in hand with a carefully tailored OPTEMPO/STRAC supported component, offers a formula to allow higher levels of training readiness across the Army than known previously.

ABOUT THE AUTHORS

MG(Ret) Bob Sunell was the first Director of the Armored Family of Vehicles Task Force retiring from that position on 31 August 1987. In addition to this critical post, he has held a wide variety of important command and staff assignments. Key among these were Program Manager for Tank Systems with the U.S. Army Tank-Automotive Command; Deputy Project Manager, XM-1 Tank Systems; Commander, 11th Armored Cavalry Regiment, U.S. Army Europe; and Commanding General, U.S. Army Training Support Center, Fort Eustis. MG Sunell holds a Bachelor's Degree from the University of Nebraska, a Master's of Science in Communications from Shippensburg State College, and is a graduate of the Marine Corps Command and General Staff College, and the Army War College. He is one of the foremost authorities on the development of combat vehicle systems and their training subsystems.

MAJ(P) Tom Rozman is currently a staff officer assigned to the Combat Developments Section of the Armored Family of Vehicles Task Force. Prior to this assignment, he served as Chief, G-3 Training Resources, 1st Armored Division, U.S. Army Europe; Executive Officer, 1st Battalion (Mech), 46th Infantry, and 2d Battalion (Mech), 6th Infantry; and Commander, Company A, 1st Battalion (Mech), 58th Infantry. MAJ Rozman is a 1970 graduate of USMA, and holds an MBA from the University of Massachusetts. He is a 1983 graduate of the Army Command and General Staff College.

TRAINING SYSTEMS: THE CRITICAL ADVANTAGE
FOR THE ARMY RESERVE COMPONENTS

Steven L. Funk
Chief, Training Support Team
National Guard Bureau
Falls Church, VA

ABSTRACT

Since the advent of the Total Force policy, the Army Reserve Components (Army National Guard and U.S. Army Reserve) have become a prime potential beneficiary of current and future technology applied to training. The Total Force policy has significantly reduced the mobilization time for the Reserve Components while placing those forces in a combat environment of rapidly increasing intensity. These conditions have converted the Reserve Components from a reserve army to an army in reserve. In spite of the similarity in mission between the Active Component and the Reserve Components, the Reserve Component training environment little resembles that of the Active Component and is little understood by the Active Component or industry. The Reserve Components' widely dispersed and constrained in training time, terrain, facilities, and equipment. Technology offers the potential to overcome many of these training difficulties. However, for the Reserve Components to benefit from technological potential, both the Active Component proponents and industry must educate themselves about the uniqueness of the Reserve Component training environment and commit to new development and marketing strategies.

INTRODUCTION

Since the advent of the Total Force Policy in 1971, Reserve Component training requirements have increased in number, complexity, and intensity. The Total Force Policy has resulted in Army National Guard (ARNG) and U.S. Army Reserve Component units with similar missions. In addition, technological advances in warfare have increased the intensity of warfare and world politics have significantly reduced the available time to mobilize. For example, the Arab-Israeli was of 1973, experienced levels of conventional destructiveness previously only associated with nuclear warfare and improvements in weapons system mobility, speed range, and lethality has moved inexorably forward. Over 75% of ARNG units must now mobilize within 60 days of notification, compared with over 120 days only a few short years ago.

A recent Army Training Board (1987) study stated that while optimizing the effect of training is the goal of every Army unit, nowhere is the mandate to do so, or the consequence of failing, more evident that in our reserve forces. The capacity of Reserve Component units to recover from even minor false starts, disconnects, and interruptions is limited by the absence of most of the inherent training flexibility available to the Active Component. Almost everything about the Reserve Component training environment is at least somewhat, and often significantly different from that of the Active Component. While the similarities between the two parts of the Total Force are important, it is the differences and their ramifications which are critical to optimizing training. The fundamental nature of the Reserve Component training environment is set by a number of truths subject to minor modification, but not to substantial change.

In this paper, I hope to identify some of these key differences and the ramifications for Reserve Component training and suggest ways in which technology can be applied to this environment.

THE RESERVE COMPONENT TRAINING ENVIRONMENT

The transfer of missions and increasingly sophisticated equipment to the Reserve Components has resulted in an increase in the absolute number of tasks to be learned and an increase in the level of proficiency with which these tasks must be practiced. This overall training requirement is exacerbated by reorganizations, unit level turbulence, geographical dispersion, competing requirement, and time constraints. So far, these words would probably bring node of recognition from any Active Component trainer and generate a "so what"? What makes the Reserve Component environment unique is the constraints within which these factors must be dealt and the ways in which these factors interact. The remainder of this section expands upon some of these points.

According to the Army Training Board (1987), approximately 20% of the Reserve Components were reorganized in FY 86. During the same year; unit level turbulence at E5 level and below was 37.5% for the ARNG and 48.7% for the USAR. When this shifting array of tasks and training audience are combined with the effects of geographical dispersion, time constraints, and facility constraints, one can begin to appreciate the differences between Reserve Components and Active Component.

According to the Institute for Defense Analysis (1987), the Reserve Components have 6900 battalions or separate companies/platoons at 3956 armories/reserve centers throughout the United States (ARNG has 3457 units in 2858 armories and the USAR has 3438 units in 1098 reserve centers. The average population of an ARNG armory is 148 and the average population of a USAR center is 202. These numbers do not necessarily mean that the total population of an armory or center belong to the same unit. It is not unusual for a company to be split between two or more armories. According to the Army Training Board (1987), the average battalion is dispersed over a radius of 150 miles with some extending over 300 miles. Division equivalent

headquarters rarely have all of their subordinate commands in one state and may extend over as many as 12 states. Units (read battalion/separate company) have to travel an average of 128.5 miles to get to their major equipment (e.g., tanks), 40.1 miles to a local training area, 154.2 miles to a major training area, and 149.2 miles to a training support center. Since these are averages, actual distances for any given unit can vary greatly. The training implication, if not already apparent, will be discussed later.

When considering training individuals in job skills, the task is magnified. Many individuals travel several hundred miles one-way to weekend training with some travelling up to 500 miles. Further, there is a low density of any given military specialty at any given armory/training center and even fewer experienced instructors. About 75% of the Reserve Component force needs some type of skill training (e.g., MOS or professional development). The Institute for Defense Analysis (1987) analyzed occupational specialties by site. They found that 13 of the 32 (41%) career management fields accounted for 33-50% of the Reserve Components. Table 1 and 2 show a representative comparison of Reserve Component skill densities per site with the Active Component for the most densely populated specialties in the Reserve Components. As you can see, the Reserve Component commander has a diverse training challenge to maintaining individual proficiency or to do skill conversion training due to reorganization or individual reassignment.

Modern weapon systems mobility, range and lethality have rendered many of the Reserve Component training areas and ranges obsolete because they require an increased amount of land for maneuver space and ranges to conduct realistic training. Adding to the land base is difficult and expensive. In some cases, the land is simply not available in the quantities needed. For example, in Iowa, less than 3% of the land is federally held and little more is state held. To add to the training land base would mean converting privately held lands. In other cases, environmental concerns make it difficult to add training lands. The net result is that units must travel greater distances to conduct realistic full scale firing and maneuver exercises.

Additionally, many units, particularly combat support and combat service support units do not have the equipment available that they would support in combat (e.g., M-1 tank maintenance units and hospital units).

To summarize all of these numbers in another way, imagine a force slightly larger than the Active Army distributed throughout the United States in mini-installations that vary in size of between 148 and 202 soldiers, very much like kasernes in Germany. Further imagine that the bulk of your equipment and training areas are one and a half hours away and that you have 10% of the potential time to train per month as an active unit during most of the year (50% during annual training). Of that 10%, the Army Training Board (1987) reports that unit commanders estimate that they can spend about one-third of their available weekends (for weekends), and about one-half (one week), of their annual training.

Of course, all of these requirements are accompanied by a concomitant increase in the number and variety of administrative tasks which must be performed by units. In fact, 71% of the unit commanders in the Army Training Board study (1987) identified some form of administration as their "real" number on priority. Less than 50% listed training among their top three "real" priorities.

The net result is that the company commander's plate is too full and complex. In addition to the increasing number and complexity of the tasks to be performed, there are physical, demographic, and time factors which constrain using the options used by the Active Component. These conditions have already impacted training effectiveness. According to the Army Training Board (1987), 62% of the company commanders report that they do not have the time to personally supervise training. They estimate that they can perform about 50% of their Army Training and Evaluation Program tasks and can sustain about 60% of the individual skills prescribed in the Soldier's Manual. To put in Active Component terms, one Reserve Component training day is the equivalent of five days for the Active Component. That means that when we force a Reserve Component unit to travel 1-1/2 hours one-way to obtain training aids or 2 hours to a local training area that is the same as forcing an Active Component unit to travel 7.5 hours to get training aids, or 10 hours for local training each way. These expectations would usually be considered intolerable in the Active force which has more time available.

All of the discussion to this point sounds pretty bleak only because it focuses on an objective status and how to do the training job better. If one were to compare the training situation in the Reserve Component forces now with as little as five years ago, tremendous progress has been made. The nature of the modern battlefield, however will not allow us to look back, but forces us to look forward. We simply must improve Reserve Component levels of performance, and we must do it within the "fundamental truths: that are not amenable to substantial change.

As a total force, we must restrict the missions to the Reserve Component units to those that are most likely to be encountered and/or are most critical to war plans and stabilize those missions over time. We need to develop training multipliers, ways to improve the leverage of the training opportunities that do exist. This requires more than a modification of the way we now do business, it requires a fundamental reorientation in the training device development and procurement arena. It requires developing training devices, simulations, and training strategies specifically based upon an analysis of Reserve Component mission tasks and environment.

RESERVE COMPONENT TRAINING DEVICE, SIMULATION, AND COURSEWARE REQUIREMENTS CHARACTERISTICS

Technology can help the Reserve Componente overcome some of the training hurdlee faced. In this eection, I point out eome of the general coneiderations and some of the physical and functional characteristice needed for training devicee, simulations, and courseware deeigned for the Reserve Componente.

Some of the general coneiderations for training in the Reserve Component environment are:

a. Take the training to the troope, it ie generally unacceptable to take troops to the training, travel time is predominantly wasted time. Ideally, individual training would be moved to the eoldier's home and low level collective training would be conducted at the armory/local training area. Field opportunities ehould provide for the most realistic training possible at the highest level of organization that the terrain will eupport.

b. Bring the outside in. That ie, bring the field into the armory or home. The extent poeeible, embed training in realietic ecenarioe which allow for escape, elow down, or replay for remediation. There is ineufficient time to routinely conduct training in linear stepe, eome training must occur by "osmosis" and build intuition or "field emarte" by operating in the actual environment or high fidelity eurrogate environment.

c. Training devices, etc., do not replace field training, we must etill go to the field, but we muet make those rare field opportunities more productive. We muet reduce the use of troops as training aide, individual/leader ekills muet be learned to the extent poseible before going to the field.

Some of the specific deeirable physical characteristice are:

a. Portability. Devicee, eimulatione, and courseware must at a minimum be deliverable in the armory/training center. Ideally, devicee could be taken to annual training with the unit and significant portions of individually oriented training would be transportable to the eoldier;s home. Devicee must be able to be put away when not in uee. Armories/training centers typically do not have enough epace to permit permanent fixtures, epace must be multipurpose.

b. Reliability. Device use ie characterized by infrequent, but intense uee by a wide variety of ueere with widely varying levele of ekill. Hence, ehipping containere muet be rugged, devices must have handles and grips that permit easy movement, and must be very forgiving to environmental variance (e.g., outside devicee will be eubject to duet, temperature extremes, and humidity).

c. Inexpeneive. When one coneiders that there are 4000 training eitee, the feaeibility of a eolution must be eeneitive to coets. For example, for trainere designed to be iseued on per armory/training center, I would coneider about $100,000 per device to be the ceiling coet to remain viable. For devices deeigned to be worketatione or eigned out to individuale to be ueed at home, $10,000 is probably about the ceiling. These are procurement coet eetimates and are not abeolutee, but are intuitive eetimates baeed upon experience over the last year. The lower the coet, the higher the probability of acceptance.

d. Computer baeed devicee ehould uee EIDS to the maximum extent poseible.

Some functional coneideratione include:

a. Armory training ehould be ecenario baeed with eecapee and elow downe/replaye for remediation. Ae previouely mentioned, time doee not permit a purely linear learning etrategy.

b. Trainers and coureeware ehould be interactive. The training ehould reepond to the actione of the trainee/crew and ehow the consequencee of their actione.

c. Scenarios ehould be realietic. The feedback ehould reflect realietic odde of eucceee, that ie the "correct" decieion in combat ecenarioe do not alwaye work, it juet improves the odde of eucceas.

d. Scenarioe should provide for variety. Reeerve Component eoldiers may move among unite, but at least in the ARNG they tend to etay in the eervice. Soldiers rapidly learn the "rulee of the game" and etop using devicee when they become repetitive. New ecenarioe and more complex scenarioe need to be routinely developed. The devices must accommodate a wide range of ekill levele from expert to remedial.

e. Devicee muet require little or no training to operate. There ie little time to perform miseion training now, there is no time for learning to uee devicee. Likewiee, devicee ehould not require dedicated instructor/operators. They must be user friendly with built-in tutoriale. Ae a guideline, the maximum time to learn how to uee a device ehould not exceed a four hour drill period.

f. Devicee need to be deeigned for uee at local training areae (e.g., MILES) that permit realietic field training to be conducted over reduced acreage. Ideally, field devicee ehould permit interplay of direct firee, indirect fires, and air eupport. Devices ehould require minimum installation and tear-down time.

Theee general parameters ehould give both proponents and contractore some idea about how to develop technologically based iteme applicable to the Reserve Component environment. Some current and past examplee of technological applicatione to thie environment are:

a. PM-TRADE is developing GUARDFIST-I and II which uses videodisk technology and EIDS to train artillery forward observers and tank crews in procedural skills in the armory using combat scenarios. GUARDFIST-I will strap on to a tank, use all of the tank controls, and involve the entire crew in video scenarios. GUARDFIST-II will allow a forward observer to practice calling fire missions by interfacing with the device in the stand alone mode or exercise the entire fire support team.

b. The Training Technology Field Activity-Gowen Field is working on two individual computer based courses. On involves conducting the Armor Basic NCO Course (BNCOC) using lap to take home computers, EIDS, and teleconferencing through USAR schools. The second project involves using computer based combat scenarios to teach tactical employment and planning skills to armor leaders at company level and below before going to the field. It amounts to a tactical exercise without troops or terrain,

c. The Army Research Institute is completing testing the use of computer based instruction as a surrogate for maintenance personnel who do not have access to end items during drill weekends.

d. The National Guard Bureau has funded initiatives for testing or developing devices amenable to armory training for the TOW and Dragon weapons systems, short range air defense weapons, and infantry squad employment. Additionally, NGB has funded additional MILES equipment for the Reserve Components and purchased devices for Regional Training Centers for Maintenance using Congressionally dedicated funds for such purposes.

The most promising technologies currently appear to be:

a. Interactive video for simulation and instructional courseware. Videodisk and various versions of CD/ROM appear applicable.

b. Ineractive courseware is a critical need in the Reserve Components. Well designed interactive courseware that encompasses most of, or complete courses are needed (e.g., NCO and officer professional development courses and MOS qualification courses).

c. Telecommunications applications that permit delivery to remote sites.

d. Surrogates for live firing in both armory and field settings such as the precision gunnery training (PGS) devices being developed by PM-TRADE for tank and Bradley training.

e. Micro-command Post Exercises where newly appointed battle staff members can practice their skills by interacting with the computer which will play the other staff officers, scenario, and provide evaluation and seminar capabilities.

STRATEGIES FOR ACTION

Reserve Component training device needs do not currently compete very successfully for either attention or funding. Nearly all Reserve Component unique new starts have been initiated through Congressionally dedicated funds for that purpose. Actions by both the military and industrial establishment can affect change in this regard. It is time to make these adjustments. As of 1987, the Reserve Components are the majority of the Total Force (52%). Further, the difference between first to fight and last to fight is becoming less significant. That difference for the ARNG for example, amounts to about 60 days.

Since there will never be enough resources to meet the continuing needs of the Active Component, placing the Reserve Component needs after the Active Component means that no significant progress will be made in the Reserve Component arena. Approaching Reserve Component training as a postscript or extension to Active Component training strategies will not result in satisfactory solutions. It is time to recognize that Army doctrine will not change American culture not the fundamental environment in which the Reserve Component soldier must operate. I recommend that the active establishment consider taking the following actions:

a. Dedicate TRADOC resources to working exclusively in the Reserve Component environment. These resources would be dedicated and not be able to be bumped by "higher" priority Active Component tasks. These resources should include research and development, training developers, procurement funds, and combat developers.

b. Fence a proportion of the Nonsystems Training Devices budget dedicated to Reserve Components. These funds would be used like the Congressionally dedicated funds of FY 86 and FY 87.

c. Dedicate a portion of exploratory research to further define the differences between Reserve Component and Active Component training. Develop approaches to address these differences.

d. Redefine the concept of "cost effective" for the training devices for the Reserve Components. The dominant current concept is the amount of use for the device per dollar (e.g., hours per dollar). Cost effectiveness should be the extent to which the device contributes to increased training readiness of the using unit. Effectiveness is not efficiency.

The current system does not sufficiently recognize a Reserve Component unique environment and results in devices that are centralized, expensive, and carry high overhead. The alternative is for the Reserve Components to continue to appeal to Congress through their respective associations for dedicated funds,

which inevitable are removed from defense line and given to the Reserve Componente by Congress. This would appear to be less deeirable that developing programs that the Reserve Components and their associatione can fully support.

Industry can also be helpful in this arena. As you can see, the Reeerve Components make up a coneiderable market that exiets in nearly all Congressional dietricte. Recognizing Reeerve Component implications when making proposale can help eensitize the whole system. Your military coneultants need to be seneitive to the differences and as they move through the Pentagon and other halls of influence to begin to sell the Reserve Component eide of the Total Force. Frequently your interests can be coordinated with ours and moved through more avenues of action than the bureaucracy alone.

SUMMARY

The Reeerve Components repreeent a unique training environment. It ie characterized by wide geographical dispersion, low training deneities at any one location, compressed training time, lack of equipment, and frequently undersized facilitiee. The elements that make this environment unique are not amenable to eubetantial change. Technology can contribute to mitigating the effects of thie environment through the development of devices and simulations that expand the number of training opportunities (e.g., practices per hour), increase the realism of training, are portable (takes training to the troope), require low overhead (e.g., dedicated operators, time to learn to operate), are interactive (e.g., provides feedback and remediation), and are inexpensive.

The Reserve Component training needs must be addressed as a unique environment by both Active Component and industry. The Reserve Components repreeent over half of the force and a eizable market. Thie market can be penetrated through the actions to the Active Component proponente, political actions of the associatione interested in Reserve Component readiness, and/or industrial marketing strategiee.

REFERENCES

Army Training Board, Enhancing Reserve Component Unit Training. Fort Monroe, VA. May 1987.

Orlansky, J.; Metzko, J.; Knapp, M.; Domin, J.; Angier, B.; Papay, J. Army Reeerve Component Training Technology-A Progrese Report. Institute for Defense Analysee. Alexandria, VA. January 1987.

ABOUT THE AUTHOR

LTC Funk is the Chief of Training Support Branch, National Guard Bureau. He holde a doctorate in education and an MBA. He has 20 yeare of active and National Guard experience and eerved as senior scientiet for two training defense contractors prior to entering full time National Guard duty.

USACMLS TRAINING DEVICES AND SIMULATIONS PROGRAM: TRAINING FOR THE NBC BATTLEFIELD'S TEMPO, SCOPE, AND UNCERTAINTY

By Cpt. William R. Bowlin
Mr. Thomas F. Carroll
Mr. Lee H. Hayslette

Training Devices and Simulations Branch
Directorate of Training and Doctrine
U.S. Army Chemical School, Fort McClellan, Alabama

ABSTRACT

The U.S. Army Chemical School (USACMLS) has initiated an innovative training device development program designed to revolutionize nuclear, biological, and chemical (NBC) training at both institutional and field levels. This task-/mission-oriented training device program trains soldiers with cues (signals designed to trigger specific actions or reactions) expected to be experienced on actual NBC battlefields, and provides realistic simulation in an area of training heretofore neglected because of troop and environmental safety constraints.

"Unit training should simulate as closely as possible the battlefield's tempo, scope, and uncertainty."

FM 100-5, Operations, 5 May 86

INTRODUCTION

Past NBC training separated NBC skills into separate training exercises--NBC olympics, or NBC volleyball or softball games played by soldiers wearing MOPP (protective) gear and gas masks. This type of training in a garrison environment appears to fulfill NBC training requirements. And training in a field environment for an Army Training and Evaluation Program (ARTEP) exercise, for instance, generally relegates NBC training to "simulation" (and the word is used lightly).

Such simulation usually requires an NBC evaluator to toss a colored smoke grenade, activate a unit's chemical agent alarm, and announce to a group of soldiers that a random selection of individuals (selected on a scientific "hey you" basis) are determined to be casualties.

It does not matter whether the individual soldier reacts properly or whether his equipment is operational. The evaluator needs casualties, so he can observe noncasualty reactions to a chemical agent attack. But soldiers are reacting to information provided by the evaluator, not information provided by the unit's organic detection equipment.

An alternative "simulation" uses riot control agents on military reservations where the public will not be harmed. The use of riot control agents trains soldiers to wear their masks when they smell CS gas (not the usual way Threat delivers chemical agents). Soldiers learn quickly who in their unit controls the use of training gas. Any time they see a known Chemical Corps soldier in a CS free play training area, they react to the presence of that individual, not to a chemical agent threat. The only alternative to these two scenarios is the introduction of training devices for simulated agent delivery and detection on a limited basis--such as the Simulated Projectile, Airburst Liquid (SPAL), simulated persistent agents polyethylene glycol 200 (PEG 200), and N-butyl-mercaptan (BUSH).

All of these systems rely on technologies borrowed from other countries and developed separately without the benefit of a unified system. They are quick fixes to individual shortfalls in training. The SPAL cannot be fired over troops, and PEG 200 simulates only one type of persistent agent, thereby making identification easy in exercises using the NBC Warning and Reporting System. BUSH gives false cues and is difficult to store. Other programs were initiated and their development and fielding continues. But now they will be a part of the USACMLS program, which began in 1982.

TRAINING DEVICE ACQUISITION STRATEGY

The mission of the Chemical School's Training Devices and Simulations (TDAS) Branch, Unit Training Division, Directorate of Training and Doctrine, is one of the most important in the Army today. One part of the TDAS story tells how this mission is being carried out. The other part explains how the program attempta to duplicate--in NBC warfare training--what the Multiple Integrated Laser Engagement System (MILES) does for direct fire, force-on-force tactical engagement simulation exercises.

This program began as an initiative of the USACMLS Commandant in June of 1982. During that month, the directors of USACMLS met to address the Army's need for NBC training devices. The Army disbanded the Chemical School in 1974 and reactivated it in 1979. During this five-year period, with no single agency assigned to coordinate NBC training technology, multiple agencies within the Army took up the tasks.

The Chemical Corps, reactivated and relocated to Fort McClellan in 1979, faced an expanded worldwide threat, antiquated equipment, and a new, dynamic doctrine for how the Army intended to fight the next war. Soldiers fighting on the AirLand battlefield must train as they are expected to fight. Playing volleyball in protective clothing and gas masks is not how the war is expected to be fought.

The result of this 1982 meeting is a document called Training Device Acquisition Strategy (TDAS). The TDAS summarizes the necessary concepts to support training in a simulated NBC environment. This document serves as the principal instrument directing efforts to conceptualize, develop, and acquire training equipment, simulants, and support packages/procedures for training.

The original TDAS consists of 22 proposed devices or systems and today remains the basis of USACMLS Training Device Acquisition Strategy Management Plan (see figure 1). As a dynamic document, TDAS continues to guide development efforts, expanding as new training deficiencies are identified, or reorganizing as developing systems are consolidated to solve multiple training deficiencies. This program supports the U.S. Army's Training and Doctrine Command (TRADOC) goals to achieve substitution, simulation, and miniaturization where possible. In the field of NBC warfare, simulation is necessary owing to the very nature of the environment in which soldiers are required to train.

TDAS Management Plan

- Nuclear Weapons Effects Simulator
- Total Dose/Dose Rate Simulator
- Radiation Automatic Casualty Assessment System
- Biological Agent Simulant
- Biological Agent Casualty Assessment System
- Biological Agent Decontamination Simulant
- Chemical and Biological Agent Delivery System
- Nonpersistent Chemical Agent Simulant
- Chemical Agent Casualty Assessment System
- Persistent Chemical Agent Simulant/Chemical Agent Disclosure Solution
- Biological Detection and Alarm Training System
- Chemical Detection and Alarm Training System
- NBC Evaluation/Training System
- NBC Computer Assisted Training System
- Scale Model NBC Equipment
- NBC Wargames and Simulations Training System
- Projected Smoke Simulator
- Infrared Defeating Smoke Simulator
- Multi-Media Threat Training System

Figure 1.

The creation of a training environment must address both system and nonsystem training devices. A system device is developed to support a specific materiel system and is designed for use only with that system. A nonsystem device supports general military training. To create a comprehensive NBC-integrated simulated battlefield, both system and nonsystem devices will be used with development carefully coordinated for maximum use and minimum cost. Therefore, the system and nonsystem training devices must work together. The development of the Chemical Agent Monitor (CAM) serves as an example of a new system development for the U.S. Army. The liquid simulant for training should be compatible with currently fielded detection paper and should be used in the field in the same way as the chemical agent it will replicate. If other detectors for the field use a simulant, the new system training simulant should be compatible.

FIELD TRAINING INITIATIVES

To coordinate USACMLS efforts with artillery and mines training systems for tactical engagement, force-on-force field training exercises, TRADOC headquarters directed its tactical engagements simulations office to monitor the Simulated Area Weapons Effects (SAWE) program. The SAWE program is a force-on-force tactical engagement system for indirect fire weapons and mines. SAWE is designed to be interoperable with the MILES system, to better replicate the total effects of the AirLand battlefield.

USACMLS contributed several components from the original TDAS document (known as SAWE-NBC) to the SAWE program and is moving ahead of development of artillery and mines components.

SAWE-NBC evolved from TDAS efforts to better define requirements for the TDAS items and to determine how they should be employed for training throughout the Army. After determining which tasks must be trained--from the soldiers manuals of common tasks (STP 21-1-SMCT and STP 21-24-SMCT)--and military occupational specialty (MOS) manuals--the efforts for training device development were redirected toward finding the appropriate "cues" to trigger actions resulting in task evaluations of both individuals and units.

The initial determination of what cues needed to be simulated for the NBC environment were made by Jet Propulsion Laboratory (JPL) in their Best Technological Approach (BTA) published May of 1986. Both the training developer community and the materiel developer met regularly with JPL to provide user guidance in development of the BTA. The Training Advisory Group (TAG) (see figure 2) provided guidance.

NBC Training Advisory Group (TAG)

• Chemical School	• Armor School
• CRDEC	• Artillery School
• PM TRADE	• British Exchange Officer
• USA Tng Sup Ctr	• German Liaison Officer
• DA Surgeon General	• FORSCOM
• USAF	• USMC
• Infantry School	• DCSOPS
• 7th ATC	• NTC

Figure 2.

The JPL BTA lists the technologies available to provide necessary cues for: initiating soldier reaction, creating the desired training environment, and assessing casualties (penalties) for inappropriate responses. The use of two technologies for three TDAS items exemplifies this. The recommended technology for the Nuclear Weapons Effects Simulator (NWES) is pyrotechnics.

The JPL prototype, which fulfills the requirements outlined by the TAG, provides visual and acoustical cues (flash and bang) of a nuclear event. The prototype NWES provides a cloud, or visual cue, and a bang, which allows soldiers to prepare appropriate reports and weapons yield calculations. This is an example of a common task skill--reaction to a nuclear attack, and an MOS-specific skill--calculation of downwind hazards.

This does not end the training, however. Once the physical cues are provided, the monitoring and survey systems also must be cued to correspond with what the soldier has seen and heard. The recommended solution for this action is radio frequency (RF) technology. By activating a transmitter simultaneously with the audio/visual cues, radio receivers constructed to operate as fielded radiological monitoring and survey instruments should reflect accurate expected readings.

This RF recommendation from the BTA for the Total Dose/Dose Rate Simulator (TD/DRS) demonstrates the systematic approach for creating an integrated training environment. When integrated with the Radiation Automatic Casualty Assessment System (RACAS), another RF system, the TD/DRS allows commanders to conduct training with a unit's simulated organic equipment, assess casualties for inappropriate subordinate commanders' decisions, and train for an environment in a manner not possible in the past. Use of the three devices named above--the RACAS, the TD/DRS, and the NWES--provides training for 20 common soldier tasks.

Many devices in the TDAS can be combined, merging individual items of similar technology into one system. Because the Biological Agent Casualty

Assessment System (BACAS), Chemical Agent Casualty Assessment System (CACAS), Radiation Automatic Casualty Assessment System (RACAS), and the Nonpersistent Chemical Agent Simulant (NCAS) are all RF technology, a single system named the NBC Casualty Assessment System (NBC-CAS) can replace all four of the single devices. This reduces the number of transmitters necessary to create the simulated NBC environment by providing one transmitter that operates on a single frequency that emits coded messages for appropriate receivers.

INSTITUTIONAL TDAS

While USACMLS moves forward to field devices for all soldiers in the area of tactical engagement field exercises, efforts continue to modernize training for resident soldiers at Fort McClellan. Central to this effort is the design of the Battle Simulation Complex (BSC) and the integration of fielded training devices and institutional elements of the TDAS. The planned facilities will allow individual and collective training using state-of-the-art technology for all ranks and services trained at USACMLS. It is also planned for use by Reserve and National Guard units. Once completed, the BSC will be capable of conducting small unit instrumented field evaluations and command post exercises for resident classes and Guard and Reserve units.

CONCLUSION

The task/mission approach to training device development shows great promise for future training Army wide. Problematic "bugs" still exist in the system, and there is room for improvement in the area of coordinating system devices for the training arena. Meanwhile, TDAS efforts to improve fielding continue, with the realization that these problems affect not just the Chemical Corps or the Army, but the entire Department of Defense. Flexibility and adaptability in the areas of coordination and openness to recommendations are the chief hallmarks of this successful problem-solving approach.

ABOUT THE AUTHORS

Cpt. William R. Bowlin is a training systems management specialist for the Training Devices and Simulations Branch, Unit Training Division, Directorate of Training and Doctrine, of the U.S. Army Chemical School. He is responsible for development of new institutional training systems and integration of NBC/smoke and obscurants into new wargames and simulations. He holds a Bachelors Degree in history from the University of Alabama in Birmingham where he is a Masters Degree candidate. Since being commissioned in 1980, CPT Bowlin has served as assistant S3/chemical officer in an infantry battalion and as platoon leader and operations officer for a chemical defense company in West Germany.

Mr. Thomas F. Carroll is the branch chief of Training Devices and Simulations Branch. He is responsible for all nonsystem training devices. Mr. Carroll has 25 years experience in the NBC field. He is retired from the U.S. Army.

Mr. Lee H. Hayslette is a training specialist in the Training Devices and Simulations Branch. He holds degrees in business administration and supervision management technology. Mr. Hayslette is retired from the U.S. Army.

VISION MOTION-INDUCED SICKNESS IN NAVY FLIGHT SIMULATORS: GUIDELINES

LCDR Michael G. Lilienthal
Naval Training Systems Center
Orlando, Florida 32813-7100

Robert S. Kennedy, Ph.D.
Kevin Berbaum, Ph.D.
Essex Corporation
Orlando, Florida 32803

LT James Hooper
Naval Air Systems Command
Washington, DC 20361-1205

ABSTRACT

Since 1982 the Naval Training Systems Center, with support from the Office of Naval Technology and the Naval Air Systems Command, has been investigating the occurrence of simulator sickness in Navy flight simulators. Simulator sickness is defined as that group of symptoms experienced by crew members in connection with maneuvers in flight simulators which would not be experienced by those same crew members in aircraft. Symptoms include nausea, dizziness, general discomfort, eyestrain, headache, disequilibrium, and pallor. In rare cases there have been aftereffects (e.g., visual flashbacks and disorientation) that have occurred up to 12 hours after exposure to the simulators. This paper reviews results of the Navy's research and subsequent recommendations that have been provided to Navy trainer acquisition managers who formulate specifications for future flight simulators. The rationale for these suggestions is derived from literature reviews, field observations, laboratory experimentation, and the results of a biomedical panel convened to address the simulator sickness problem.

Background

Training, the military's primary mission during peacetime, has large and continuing demands on the financial resources allocated to the Department of Defense. For example, it costs about $3.6 billion per year for fuel and supplies needed to operate military aircraft in the United States. Many of these operations are conducted for training purposes. Flight simulators, however, can be operated at costs that vary from 5%-20% of the cost to operate comparable aircraft [20]. Generally, pilots trained in simulators can acquire necessary mission skills with fewer flight hours than those pilots who are not trained in simulators.

Advancing engineering technologies permit a range of capabilities to simulate the real world through very compelling kinematics and computer generated visual scenes. This synthetic vestibular and visual environment can, on occasion, be so successful that conflict is established among the different forms of sensory input. This discrepancy adheres most closely to the cue conflict theory of motion sickness which has been accepted as the working model for a phenomenon labeled as simulator sickness [10]. In brief, the model postulates the referencing of motion information signaled by the retina, vestibular apparatus or proprioception to "expected" values based on a neural store which reflects past experience. A conflict between expected and experienced flight dynamics of sufficient magnitude can exceed a pilot's ability to adapt, inducing in some cases simulator sickness [10].

Simulator sickness is considered to be a form of motion sickness. Motion sickness is a general term for the constellation of symptoms which result from exposure to changing accelerations, but occasionally changing visual motions may also induce the malady. Motion sickness is characterized by vomiting and retching, and has overt signs of pallor, sweating, salivation, drowsiness, and nausea [12].

The symptoms of simulator sickness found in Table 1 parallel those of motion sickness [12, 19] and can produce serious residual aftereffects in pilots [6, 17]. Differences between the symptoms of simulator sickness and more common forms of motion sickness are that in simulator sickness, visual symptoms tend to predominate and vomiting is rare. The aftereffects already pose a threat to operational readiness because, in some simulators, pilots are restricted in their activities after simulator hops.

Navy Programmed Study of Simulator Sickness

For the past five years, the Navy has conducted a program of research on simulator sickness. This program was initiated to (1) provide problem definition using field survey data [6, 11, 13, 14], (2) conduct a review of the literature [3, 4, 5, 12], and (3) convene a workshop [15].

Results from the 10 Navy flight simulators that have been surveyed show a variation in their incidence from 10%-60% [11]. Table 2 provides the results by simulator. Table 3 shows the self-reported incidence of four characteristic symptoms of motion sickness (which are also characteristic of simulator sickness) -- dizziness with eyes open, vertigo, stomach awareness, and nausea for each of the 10 simulators. The samples for each symptom exclude instances in which symptoms occurred prior to simulator exposure.

Table 4 presents for each simulator the self-reported incidence and frequency of eyestrain symptoms -- headache, eyestrain, and difficulty focusing. These symptoms are less likely than motion sickness symptoms to habituate during training. Again, pilots employed were those who were free of symptoms upon entering the simulator. From these tables it is clear that some simulators elicit symptoms in few individuals, whereas other simulators elicit symptoms in many.

TABLE 1. DIAGNOSTIC CATEGORIZATION OF SIMULATOR SICKNESS

PATHOGENIC SYMPTOM
- Vomit

MAJOR SYMPTOMS
- Increased salivation
- Nausea
- Sweating
- Pallor
- Retch
- Drowsiness

MINOR SYMPTOMS
- Increased salivation
- Nausea
- Pallor
- Sweating
- Drowsiness

MENTAL SYMPTOMS ("minor" and "other" symptoms)
- Difficulty concentrating (minor symptom)
- Confusion (minor symptom)
- Fullness of head (other symptom)
- Depression (other symptom)
- Apathy (other symptom)

VISUAL SYMPTOMS ("minor" and "other" symptoms)
- Difficulty focusing (minor symptom)
- Visual flashbacks (minor symptom)
- Blurred vision (other symptom)
- Eye strain (other symptom)

"OTHER SYMPTOMS"
- Character facies
- Increased yawning
- Stomach awareness
- Anorexia
- Burping
- BM desire
- Headache
- Dizziness
- Aerophagia
- Vertigo
- General fatigue
- Experimenter's report of emesis

TABLE 2. INCIDENCE OF SIMULATOR SICKNESS SYMPTOMATOLOGY

SIM-ULATOR	N	AIRCRAFT	SIM-TYPE	LOCATION	INCIDENCE*
2E7	94	F/A-18	WTT	Lemoore	31%
2F132	26	F/A-18	OFT	Lemoore	27%
2F112	52	F-14	WST	Miramar	10%
2F110	55	E-2C	OFT	Miramar	47%
2F64C	223	SH-3	WST	Jacksonville	60%
2F87F	66	P-3C	WST	Jax/Brunswick	39%
2F117	281	CH-46	WST	New River	26%
2F121	159	CH-53D	OFT	New River	36%
2F120	230	CH-53E	OFT	New River	33%

Total N = 1186

*NOTE: (CRITERION: At least one minor symptom checked off on the POSTHOP Symptom Checklist)

Using data from these tables, the simulators may be classified into categories of high, medium, and low symptom frequencies. (The data for the 2F120NR will be excluded since only 14 cases were available from that simulator). Tables 5 and 6 present these classifications for motion sickness and eyestrain symptoms, respectively. There is some, but not complete, agreement between the classifications of simulators according to the two symptom categories. Two simulators (i.e., 2F120T and 2F64C) produced a high incidence of both motion sickness and eyestrain, two other simulators (i.e., 2F112 and 2F132) produced low incidence of both symptom categories, and one simulator (i.e., 2F121) produced medium incidence of both categories. The other four simulators (i.e., 2F110, 2F117, 2F87F, and 2E7) had a one-level difference (high/medium or medium/low) between production of the two symptoms.

Studies of Motion Systems

A recent study examined the effects on sickness rates of differing energy spectra in moving-base simulators [1]. The 2F64C (SH-3) and 2F87F (P-3C) simulators were selected because, while both were moving-base simulators, the 2F64C previously had revealed a high incidence of simulator sickness, but the 2F87F did not. During data collection, the simulators were in constant use (15 to 16 hours/day) for military aviation training by fleet replacement pilots, operational squadron pilots, and midshipmen.

Figure 1 shows a comparison of the nominal mean run of the 2F87F simulator with the nominal run for the 2F64C simulator, overlaid on Military Standard 1472C (MIL-STD-1472C) [18] for exposure to Very Low Frequency (VLF) vibration. It is obvious that the force environment of the two devices is markedly different, and that the 2F64C presents motion profiles largely in regions which MIL-STD-1472C counsel against if one is to avoid the nauseogenic features of the VLF vibration. Furthermore, differences in the pilots' reports of sickness were statistically increased by exposures in the 2F64C but were virtually absent in the 2F87F.

These findings reveal that the incidence of sickness was greater in a simulator with energy spectra in the region described as nauseogenic by MIL-STD-1472C and high sickness rates were experienced as a function of time exceeding these VLF limits. Therefore, for any moving-base simulator reported to have high incidences of sickness, frequency x acceleration recordings of pilot/simulator interactions should be made and compared with VLF guidelines from MIL-STD-1472C. However, in those cases where illness occurs in a fixed-base simulator, other explanations must be sought. These results are dealt with more completely elsewhere [22].

RECOMMENDATIONS

Motion System Specifications

If a moving base is specified, we recommend that the accelerations at or near 0.2 Hz be avoided or at least limited to 1/4 to 1/2 of the MIL-STD-1472C VLF vibration 10% nausea limits for motion sickness. Research by McCauley and Kennedy [16] have determined that exposure to sufficient acceleration energy at or near 0.2 Hz will induce motion sickness in 10% of those exposed to that energy (see Figure 1). Most simulator sickness is self-limiting in that few pilots remain in the simulator until they vomit since milder symptoms appear (such as headache and nausea) which cause the pilot to discontinue the simulator flight long before vomiting. The previously cited comparison of two simulators (i.e., 2F87F and 2F64C) with different energy spectra (Van Hoy et al., 1987) and different reported incidence rates of simulator sickness indicate that sickness, in this case, is a function of the time the pilot spent exceeding the VLF limits found in MIL-STD-1472C. However, this finding does not mean that exceeding the limits is the only causal factor of simulator sickness. The occurrence of sickness in fixed-based systems [21] is of particular interest because of the absence of inertial displacements. This strongly indicates that unknown characteristics of optical information which normally accompany and visually specify inertial displacements may lead to illness in some simulators.

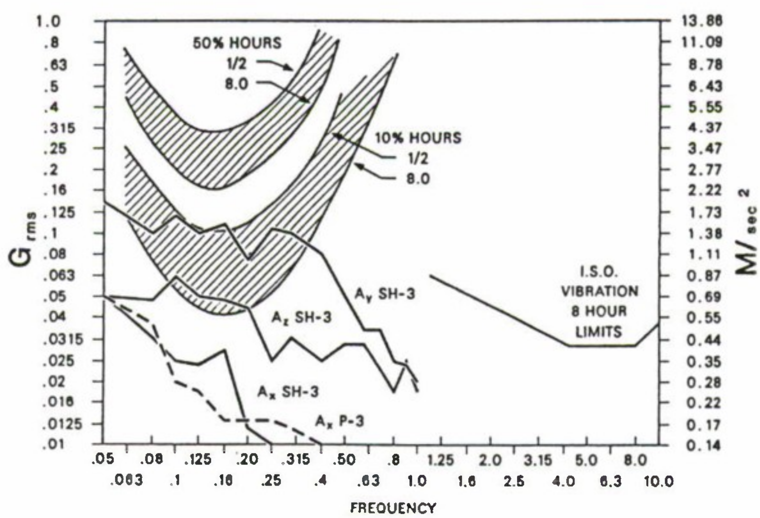


Figure 1. SH-3 (2F64C) Sea King nominal energy spectra frequency analysis versus P-3C (2F87F) Ax (x AXIS MOTION)

TABLE 3. PRIMARY MOTION SICKNESS SYMPTOMS

(a) Percentages of those not reporting a symptom before exposure that report the symptom after exposure)

Simulator	Dizziness	Vertigo	Stomach Awareness	Nausea
2E7	4.2%	10.5%	12.2%	5.4%
2F132	0.0%	0.0%	0.0%	0.0%
2F112	0.0%	2.9%	6.1%	0.0%
2F110	7.6%	5.7%	3.8%	5.7%
2F64C	6.3%	4.2%	14.1%	13.3%
2F87F	4.8%	0.0%	0.0%	0.0%
2F117	2.0%	2.7%	10.2%	8.9%
2F121	0.6%	2.6%	4.0%	6.5%
2F120NR*	0.0%	7.1%	0.0%	7.7%
2F120T*	6.7%	8.3%	6.7%	13.6%

(b) Ratio of Frequencies (Frequency of postexposure symptoms report with preexposure negative report/Frequency of preexposure negative report of symptoms)

Simulator	N	Dizziness	Vertigo	Stomach Awareness	Nausea
2E7	95	4/95	10/95	11/90	5/92
2F132	22	0/22	0/22	0/22	0/22
2F112	34	0/34	1/34	2/33	0/34
2F110	53	4/53	3/53	2/53	3/53
2F64C	144	9/144	6/144	20/142	19/143
2F87F	21	1/21	0/21	0/21	0/20
2F117	148	3/147	4/148	15/147	13/146
2F121	156	1/156	4/156	6/156	10/153
2F120NR*	14	0/14	1/14	0/13	1/13
2F120T*	60	4/60	5/60	4/60	8/59

*A 2F120 simulator is available at MCAS New River (NR), NC and MCAS Tustin (T), CA. They are similar, but not identical.

TABLE 4. EYESTRAIN RELATED SYMPTOMS

(a) Percentages of those not reporting a symptom before exposure that reported a symptom after exposure

Simulator	Headache	Eyestrain	Difficulty Focusing
2E7	7.5%	12.6%	6.3%
2F132	0.0%	18.2%	4.8%
2F112	0.0%	0.0%	0.0%
2F110	20.0%	22.9%	9.4%
2F64C	26.9%	38.4%	23.6%
2F87F	10.5%	25.0%	9.5%
2F117	8.6%	11.7%	2.1%
2F121	9.8%	16.8%	4.6%
2F120NR*	7.1%	7.1%	0.0%
2F120T*	25.5%	29.4%	15.0%

(b) Ratio of Frequencies (Frequency of postexposure symptoms report with preexposure negative report/Frequency of preexposure negative report of symptoms)

Simulator	N	Headache	Eyestrain	Difficulty Focusing
2E7	95	7/94	11/87	6/95
2F132	22	0/20	4/22	1/21
2F112	34	0/33	0/33	0/33
2F110	53	10/50	11/48	5/53
2F64C	140	36/134	51/133	33/140
2F87F	21	2/19	5/20	2/21
2F117	148	12/140	17/145	3/145
2F121	156	15/153	25/149	7/153
2F120NR*	14	1/14	1/14	0/14
2F120T*	60	14/55	15/51	9/60

*A 2F120 simulator is available at MCAS New River, NC and MCAS Tustin, CA. They are similar but not identical.

TABLE 5. CHARACTERISTICS OF SIMULATORS THAT ELICIT MOTION SICKNESS-LIKE SYMPTOMS

Simulator	Nausea	6 DOF Motion Base	FOV H/V (Degrees)	CRT/Dome	Helo/ Fixed Wing	Image Generation
High Incidence						
2F120T	13.6%	yes	200/50 & chin window	CRT	Helo	Digital CGI/ raster CRT
2F64C	13.3%	yes	130/30 & chin window	CRT	Helo	Dusk/Night CGI/calli- graphic CRT
2F117	8.9%	yes	175/50 & chin window	CRT	Helo	Day/Night/ Dusk CGI/ raster CRT
Moderate Incidence						
2F121	6.5%	yes	200/50	CRT	Helo	Digital CGI/ raster CRT
2F110	5.7%	yes	139/35	CRT	Fixed	Dusk/Night CGI/Hybrid virtual image CRT
2E7	5.4%	no	360/145	Dome	Fixed	Day/Night/ Dusk CGI/ TV camera A/C models
Low Incidence						
2F112	0.0%	No	350/150	Dome	Fixed	TV camera A/C model point light source Background
2F132	0.0%	no	48/32	CRT	Fixed	Day/Night/Dusk CGI/CRT Raster
2F87F	0.0%	Yes	48/36	CRT	Fixed	Day/Night/Dusk CIG/off axis reflective

TABLE 6. CHARACTERISTICS OF SIMULATORS THAT ELICIT EYESTRAIN-RELATED SYMPTOMS

Simulator	Headache	6 DOF Motion Base	FOV H/V (Degrees)	CRT/ Dome	Helo/ Fixed Wing	Image Generation
High Incidence						
2F120T	25.5%	yes	200/50 & chin window	CRT	Helo	Digital CGI/ Raster CRT
2F64C	26.9%	yes	130/30 & chin window	CRT	Helo	Dusk/Night CGI/ Calligraphic CRT
2F110	20.0%	yes	139/35	CRT	Fixed	Dusk/Night CGI/ Hybrid virtual image CRT
Moderate Incidence						
2F121	9.8%	yes	200/50	CRT	Helo	Digital CGI/ raster CRT
2F87F	10.5%	yes	48/36	CRT	Fixed	Day/Night/Dusk CGI/ off axis reflective
2F117	8.6%	yes	175/50 & chin window	CRT	Helo	Day/Night/Dusk CGI/Raster CRT
Low Incidence						
2F112	0.0%	no	350/150	Dome	Fixed	TV camera A/C model point light Background source
2F132	0.0%	no	48/32	CRT	Fixed	Day/Night/Dusk CIG/CRT raster projection
2E7	7.5%	no	360/145	Dome	Fixed	Day/Night/Dusk CGI/ TV camera A/C models

Visual System Specifications

It is recognized that simulator sickness is a by-product of our high technology capability that allows a rich and compelling visual imagery that aviators find realistic and satisfying as substitutes for the real aircraft. These features and simulator sickness may need to be traded off, keeping in mind that training effectiveness is the primary criterion. Our recommendations to reduce simulator sickness must be understood in this context of cost, maintenance, and training trade offs. It is recommended that a dome projection display be used rather than multiple infinity optics CRT displays. Table 6 indicates that eyestrain symptoms such as headache are reported and observed in CRT/infinity optics simulators much more often than in domed, projection simulators. Eyestrain-related symptoms may in some cases exacerbate conditions of simulator sickness.

The optometric and ophthalmological communities' recommendations concerning the causes and remedies to reduce asthenopia (i.e., eyestrain) are found in McCauley [15]. It appears that eyestrain may be caused by the conflict among the simultaneous disparate vergence, version, and accommodation demands for the different distance cues in the visual simulator system. Accommodative-convergence would signal excessive vergence relaxation while convergence-accommodation would call for excessive (positive) accommodation. Each system would be under tension and require error correction under negative feedback control. Experts in the optometric and ophthalmological communities [8] frequent, even small, shifts in accommodation and/or convergence which satisfy the need for error correction, are sufficient conditions for the generation of eyestrain.

If a dome projection system is not possible, the design eye points should be a constant radius from the operator for all CRTs. A change in the distances of the CRTs collimating mirrors forces a constant shift in the accommodation/ convergence of the pilot's visual system as he views the visual scenes across CRTs. The collimating mirror distances from the design eye point range, for example, from 47 to 92 inches for the 2F120 helicopter simulator, and from 42 to 75 inches for the 2F64C helicopter simulator (see Table 7). These two simulators have a much higher incidence of eyestrain-induced problems than domed simulators. Besides the benefit of reduced stress on the visual system of the pilot, a projected visual system allows both crew members of a simulator access to the visual scene and thus allows for crew coordination training.

Future Research

No single factor will account for sickness in all simulators. The study of this malady is complicated by the fact that the symptoms are different in different people, in the same person on successive occasions, and in different simulators. Motion-base systems, computer-generated imagery, long hops and helicopter simulators appear to produce greater than average eyestrain, changes in ataxia, and other adverse symptoms. Since many of these conditions appear in the same simulators, therefore, it is generally agreed that no single simulator attribute has been found to be clear-cut in giving rise to more problems than any other. Future technological work must conduct converging operations on a variety of elements to determine the extent that each contribute to the incidence of simulator sickness.

TABLE 7. PILOT DESIGN EYE POINTS FOR THE 2F120 AND 2F64C SIMULATORS

CH-53E (2F120)	PILOT DESIGN EYE POINT					
	LEFT SIDE	LEFT QTR	FRONT	RIGHT QTR	RIGHT SIDE	CHIN
MIRRORS RADIUS OF CURVATURE (INCHES)	60	60	50	60	50	60
DISTANCE TO EYEPOINT	92	76	47	70	50	89
DIFFERENCE	+32	+16	-03	+10	0	+29

SH-3(2F64C)	COPILOT		BOTH		PILOT	
	DESIGN EYE POINTS					
MIRROR RADIUS OF CURVATURE	49.5	49.5	49.5	49.5	49.5	66
DISTANCE TO EYEPOINT	40*	44*	49.5	42*	46	75
DIFFERENCE	-9.5	-5.5	0	-7.5	-3.5	+9

*IBERIA OPTICS: NOMINAL EYEPOINT NORMALLY LIES APPROXIMATELY 3 INCHES FROM LINE FROM CENTER OF SPHERICAL MIRROR TO CENTER OF CURVATURE OF SPHERICAL MIRROR

A common factor shared by fixed-base simulators with high levels of reported sickness is the presence of a visual display capable of presenting the dynamic visual transformations that specify aerial self motion. Many sickness reports occur in newer simulators that possess superior visual imagery capabilities. Patterns of motion of visual elements (optical flow) ecologically valid for self movement can be displayed over a wider area with greater spatial and temporal resolution, brightness, and contrast. The problem is to identify the properties of the visual stimulus which contribute to the impression of illusory self motion (vection) and to determine whether these properties result in simulator sickness.

The term "vection", refers to a stationary observer's illusory experience of self motion induced by perceived transformations in the optic array similar to those which normally accompany physical movement through the environment. This phenomenon, which is the percept likely to be at the crux of the compelling impressions of motion which occur in simulator and cinerama, has become the topic of considerable study [7, 9]. A panel of experts provided consensus, although not unanimous, that vection seems to be a necessary though insufficient cause for the occurrence of simulator sickness [7, 2] -- sickness generally does not occur in the absence of vection, but the experience of vection does not necessarily lead to sickness.

The trend towards enhanced realism in simulation contributes to more powerful experiences of vection on the part of observers. However, the relationship between display characteristics which give rise to vection and those which tend to promote simulator sickness need to be clarified in order to provide acceptable design criteria for training devices. Moreover, the development of simulator visual systems must insure that the pilot's impression of self-movement in the simulator will not be at odds with the experiences in the actual aircraft.

ACKNOWLEDGMENT

This paper summarizes research supported by the Office of Naval Technology program element 62757N, Task No. 3775 and Naval Air Systems Command funding. The opinions expressed in this paper are those of the authors and do not necessarily represent those of the Naval Training Systems Center, the Department of the Navy, or the Department of Defense.

REFERENCES

1. Allgood, G. O., Kennedy, R. S., Van Hoy, B. H., Lilienthal, M. G., & Hooper, J. H. (1987). The effects of very-low frequency vibrations on simulator sickness. Paper presented at the 58th Annual Scientific Meeting of the Aerospace Medical Association, Las Vegas, NV.

2. Andersen, G. L., & Braunstein, M. L. (1985). Induced self-motion in central vision. Journal of Experimental Psychology, Human Perception and Performance, 11, 122-132.

3. Casali, J. G. (1986). Vehicular simulation-induced sickness: Volume I: An overview (NTSC-TR-86-010). Orlando, FL: Naval Training Systems Center. (NTIS No. AD A173-904)

4. Casali, J. G., & Wierwille, W. W. (1986a). Vehicular simulator-induced sickness. Volume II: A selected annotated bibliography (Final TR-86-011). Orlando, FL: Naval Training Systems Center. (NTIS No. AD A172-990)

5. Casali, J. G., & Wierwille, W. W. (1986b). Vehicular simulator-induced sickness. Volume III: Survey of etiological factors and research facility requirements (Final Report TR-86-012). Orlando, FL: Naval Training Systems Center. (NTIS No. AD A173-226)

6. Crosby, T. N., & Kennedy, R. S. (1982). Postural disequilibrium and simulator sickness following flights in a P3-C operational flight trainer. Preprints of the 53rd Annual Scientific Meeting of the Aerospace Medical Association, Bal Harbour, FL.

7. Dichgans, J., & Brandt, T. (1978). Visual-vestibular interaction: Effects on self-motion perception and postural control. In R. Held, H.W. Leibowitz, & H.L. Teuber (Eds.), Handbook of sensory physiology: Vol. 13. Perception. Berlin: Springer-Verlag.

8. Ebenholtz, S. M. (1986) Properties of adaptive oculomotor control system and perception. Acta Psychologica, 63, 1-14.

9. Johansson, G. (1978). Visual event perception. In R. Held, H. W. Leibowitz, & H.L. Teuber (Eds.), Handbook of sensory physiology: Volume 8. Perception. Berlin: Springer-Verlag.

10. Kennedy, R. S., Berbaum, K. S. & Frank, L. H. (1984). Visual distortion: The correlation model (Technical Report No. 841595). Proceedings of the SAE Aerospace Congress & Exhibition, Long Beach, CA.

11. Kennedy, R. S., Dutton, B., Ricard, G. L., & Frank, L. H. (1984) Simulator sickness: A survey of flight simulators for the Navy (Technical Report No. 841597). Proceedings of the SAE Aerospace Congress & Exhibition, Long Beach, CA.

12. Kennedy, R. S., & Frank, L. H. (1986). A review of motion sickness with special reference to simulator sickness. Paper presented at the 65th Annual Meeting of the Transportation Research Board, Washington, DC.

13. Kennedy, R. S., Lilienthal, M. G., Dutton, B., & Ricard, G. L. (1984) Simulator sickness: Incidence of simulator aftereffects in Navy flight trainers. Proceedings of the 22nd SAFE Symposium (pp. 299-302). Las Vegas, NV.

14. Kennedy, R. S., Merkle, P. J., & Lilienthal, M. G. (1985). A comparison of postural equilibrium effects following exposure to different ground-based flight trainers. Proceedings of the SAFE Symposium, Las Vegas, NV.

15. McCauley, M. E. (Ed.). (1984). Simulator sickness: Proceedings of a workshop. National Academy of Sciences/National Research Council/National Academy of Sciences, Committee on Human Factors, Washington, DC.

16. McCauley, M. E. & Kennedy, R. S. (1976) Recommended human exposure limits for very-low-frequency vibration (Technical Publication 76-36). Point Mugu, CA: Pacific Missile Test Center.

17. McGuinness, J., Bouwman, J. H., & Forbes, J. M. (1981) Simulator sickness occurrences in the 2E6 Air Combat Maneuvering Simulator (ACMS) (NAVTRAEQUIPCEN 80-C-0135-4500-1). Orlando, Fl: Naval Training Equipment Center.

18. Military Standard 1472C. (1981, May). Human Factors engineering design criteria for military systems and facilities (MIL-STD-1472C). Washington, DC: Department of Defense.

19. Miller, J. W. & Goodson, J. E. (1960) Motion sickness in a helicopter simulator. Aerospace Medicine, 31, 204-212.

20. Orlansky, J. & String, J. (1979). Cost-effectiveness of flight simulators for military training devices. Proceedings of First Interservice and Industry Training Equipment Conference.

21. Uliano, K. C., Kennedy, R. S., & Lambert, E. Y. (1986). Asynchronous visual delays and the development of simulator sickness. Proceedings of the Human Factors Society 30th Annual Meeting (pp. 422-426). Dayton, OH: Human Factors Society.

22. Van Hoy, B. W., Allgood, G. O., Lilienthal, M. G., Kennedy, R. S., & Hooper, J. M. (1987, June). Inertial and control systems measurements of two motion-based flight simulators for evaluation of the incidence of simulator sickness (pp. 265-273). Proceedings of the IMAGE IV Conference, Phoenix, AZ.

ABOUT THE AUTHORS

LCDR MICHAEL G. LILIENTHAL received his Ph.D. in experimental psychology from the University of Notre Dame in 1978. He is a Navy Lieutenant Commander with nine years of professional experience in human factors engineering. He has a designated subspecialty of Weapon Systems Acquisition Management (WSAM) and is an Associate Fellow of the Aerospace Medical Association. He presently is the Branch Manager of the Training Technologies Development Branch, Human Factors Division, of the Naval Training Systems Center.

DR. ROBERT S. KENNEDY is Vice President of Essex Corporation and Facility Director of Essex' Orlando office. He has more than 25 years' experience in applied experimental psychology and development, and has worked extensively in the areas of vestibular and visual perception, motion and simulator sickness, and individual differences in human performance.

DR. KEVIN S. BERBAUM is a consulting scientist with Essex Corporation. He has been involved in many forms of applied vision research and is currently involved in the evaluation of area-of-interest displays and the investigation of simulator sickness.

LT JAMES HOOPER is a designated aerospace experimental psychologist stationed at the Naval Air Systems Command, APC 205. He has several years of professional experience in human factors engineering and training systems design. He currently holds a Master's degree in experimental psychology. Lt. Hooper has a vast amount of "hands-on" experience with aircraft and weapons systems from his tour of duty as a helicopter pilot during the Vietnam war.

CHALLENGES TO THE JOINT SERVICES V-22 OSPREY TOTAL TRAINING SYSTEM

Major Daniel C. Schultz, USMC, and Commander Jerry M. Owens, USN,
Naval Air Systems Command, and
Lieutenant Commander Steven D. Harris, USN,
Naval Air Test Center

ABSTRACT

The V-22 Osprey aircraft is envisioned as a versatile and complex weapon system that will offer unprecedented mission flexibility for the Army, Navy, Air Force and Marine Corps well into the next century. The V-22 will incorporate advanced technology in its composite airframe, aerodynamics, avionics, and cockpit design, and will provide the United States with a highly mission capable aircraft. Along with the system's unique flight characteristics and advanced technology underlying its construction and suite of mission systems, an expanded set of challenging new requirements for training system design is rapidly emerging. The degree to which the services and the training industry can successfully anticipate, identify and address the requirements for optimal training will largely determine the ultimate effectiveness of the V-22. This paper reviews major considerations in the V-22 training system development to date, examines some of the more salient training issues, and challenges the training systems community to develop innovative solutions that capitalize on advanced technology and maximize training effectiveness.

INTRODUCTION

The MV-22 design that was negotiated for full scale development in May 1986 will produce a vertical short take-off and landing (VSTOL) aircraft that combines the fixed wing advantages of high-speed, high-altitude (300 kts, 30,000 ft) with the helicopter advantages of confined area landings and slow speed flight. Derived predominately from American technology and ingenuity, the V-22's capability to transition from rotor-borne flight to wing-borne flight will be accomplished by tilting engine nacelles that produce vectored thrust. The airframe will be advanced composite construction, the avionics will be totally computer-controlled, and the cockpit will comprise the latest innovations in control/display technology.

In addition to building one of the most technologically advanced aircraft, the Navy, in cooperation with the Army, Air Force and Marine Corps, must develop the V-22 Osprey for joint operational use. The Marines (552 aircraft) and Army (231 aircraft) will employ the MV-22A in combat assault, combat support, service support and MEDEVAC missions. The Navy (50 aircraft) will use the HV-22A system variant for combat search and rescue, special warfare and fleet support. The Air Force (80 aircraft) will use the CV-22A variant for long range special operations and combat search and rescue. Two other variants are already on the design board: 300 SV-22s to support the Navy's anti- submarine warfare mission; and the VV-22 for the Marines in support of the presidential mission.

The V-22 is a multi-role, multi-service vehicle that poses numerous technical challenges for DOD and the defense industry in meeting the joint services' mission performance requirements. The Navy, as the executive agency for procurement of the V-22, faces the technical challenges with a firm resolve to deliver the Osprey on time and within budget. Training is of paramount concern among these challenges. In the present paper, we review major considerations that have influenced developments in the V-22 training system program to date and describe some current views on outstanding training system requirements. A major objective of the paper is to stimulate the training systems community at large to develop innovative, cost-effective approaches to realize the greatest training and mission effectiveness possible for a new generation of military aviators.

INITIAL TRAINING SYSTEM DESIGN CONSIDERATIONS AND DEVELOPMENTS

Interest in the V-22 from both weapon system developers and training system developers is easy to understand in view of the aircraft's projected multi-service and commercial roles. What is less easily grasped is the complexity of many of the design issues, especially the training system design issues. To appreciate the more salient problems and issues that have arisen in early stages of the training system design, it is necessary to expand upon the uniqueness of the Osprey's design features and mission capabilities. The capability and versatility of this aircraft have dictated novel aspects in the training system design approach to date and, moreover, leave room for considerably more innovation as we attempt to determine the full range of total system requirements.

V-22 General Characteristics

The V-22 is not an aircraft that falls conveniently into one class of system (i.e., helicopter, propellor or jet) with a unidimensional role (e.g., cargo, patrol or attack). Culminating 30 years of research and development in tiltrotor technology, the Osprey is often described as an airplane capable of landing like a helicopter, or a helicopter capable of flying like an airplane. Actually, neither of the above descriptions is adequate; the V-22 is a VSTOL aircraft that flys on vectored thrust. The transition mode from rotor-borne to wing-borne flight is referred to as the conversion corridor and is defined by a complex functional relationship between airspeed and nacelle angle. It is bounded on the low end by the stall speed of the wing, and at the high end by the aeroelastic stability requirements of the wing and nacelle as a unit. As a vectored thrust machine, the V-22 will require new piloting control skills linked to knowledge of vector geometry during transition flight phases.

Mastery of the basic piloting control skills required by the V-22 will likely represent a secondary challenge compared to the proficiency an aviator will be expected to develop in other interactions with one of the most highly advanced flight crewstations ever designed. The Cockpit Management Display System (CMDS) serves as the nerve center of the total aircraft system integration. The cockpit is described as all electronic and all glass (i.e., all information displayed via CRTs). Other features of the crewstation include: the symmetrical layout of multifunction color displays and controls for pilot and copilot access of all aircraft subsystems and instrument readouts; helmet-mounted displays for sensor (e.g., forward looking infrared system) slewing and targeting capabilities; a digital moving map display; tactical decision aids; and data links to remote tactical controllers. Flight control is accomplished through a digital fly-by-wire, triply redundant system. In the V-22, the concept of the aviator as a systems manager of a broad array of sophisticated capabilities has been fully realized.

From the brief outline of design features presented above, it is clear that the Osprey will significantly impact aviation operations across the services. The V-22 will not only replace numerous aircraft presently in the inventory (e.g., CH-46, CH-53), but also will make obsolete many of the current helicopter and fixed-wing tactics associated with the aircraft for which it can substitute. With extended range, greater speed, and greater payload capacity (2-3 times the CH-46), the V-22 will expand the entire battlefield. Its forward insertion capabilities will threaten enemy communication links, command and control, and supply lines in ways not heretofore possible. The myriad of missions that will be served by the V-22 across the services must be accomplished, however, against increasingly capable threats.

A multitude of training system design implications and strategies follows directly from the V-22's unique design characteristics and multi-mission capabilities. From the outset, it was clear that critical elements of the V-22's training system approach would differ dramatically from any aviation training system predecessor. In the next section, highlights of some of the more important developments in the training system design to date are reviewed.

Initial Developments In V-22 Training System Design

Three major issues were encountered that heavily influenced the early stages of the training system design. The first issue involved joint service agreement on tasks to be trained in operational flight trainers (OFTs) and aircrew system trainers (ASTs). Hardware to support training curricula had to be defined but it was not apparent that a single system could satisfy all the services' requirements. Differences in each of the services's views of their respective mission requirements had to be addressed and the extent of commonality between tasks to be trained by the services had to be established. The second issue concerned the entry level skill requirements of pilots to fly the V-22 and the means by which these requirements would be satisfied. An extensive, objective analysis was required to address the problem that was further confounded by widely disparate views on the V-22's characteristics as mainly an airplane or mainly a helicopter. The third issue required a plan to accomplish transition training for aviators already in the system who must be prepared to fly the first operational V-22s.

The first major issue was resolved through the Instructional Systems Development (ISD) process and efforts of subject matter experts. A pilot task listing was developed by the prime manufacturers, Bell/Boeing, drawing upon the tiltrotor flight expertise of their flight test engineers and test pilots. Due to differences in terminology and multi-service mission requirements, members of the joint service fleet project team (FPT) experienced initial difficulties in validating the accuracy of the task listing. An in-depth analysis by the joint FPT, however, determined that approximately 97% commonality existed among training tasks across the services. Following this assessment, the joint FPT reviewed and approved specifications for the development of 13 OFTs and 6 ASTs for joint service use. A modular design approach was directed for construction of the trainers to allow for pre-planned product improvement, accomodation of service unique requirements, and changes in the aircraft design. The up-front incorporation of modular design technology will allow the necessary room for variation and upgrade as well as the capability for trainers to "grow" into unique mission requirements.

Resolution of the second issue concerning student entry level skill requirements will have far-ranging impacts upon the training system design, organizations, and costs. From a practical standpoint, training conducted in one of the traditional jet, helicopter or maritime pipelines, or possibly some combination or improvement of the existing pipelines, would seem to be a cost-effective solution with the least organizational impact. Popular opinions initially supported improvements to existing pipelines but the opinions were biased by aircraft community specific experience. As another alternative, an entirely new trainer aircraft (TV-XX) could be developed to support a new curriculum.

The Naval Air Systems Command addressed the problem of student entry level skill requirements in conjunction with the Chief of Naval Education and Training (CNET), the Chief of Naval Air Training (CNATRA) and experienced contractor personnel. Generic and specific V-22 task listings were evaluated by instructor pilots from jet, helicopter and maritime pipelines with each task rated as to the degree to which associated skills were trained. Additional reference information was obtained from the same evaluation of task lists and interviews with pilots and instructors from the AV-8B, CH-46 and CH-53 communities; XV-15 pilots were also interviewed. The results indicated that none of the existing pipelines trains a majority of the V-22 piloting and mission systems tasks. Combined helicopter-jet or helicopter-maritime pipelines result in training of only slightly more than 50% of the tasks. The findings bear out a point made above:

the V-22 cannot be classified as either a helicopter or an airplane. It shares the aerodynamic qualities of helicopters, AV-8B's and conventional turboprop aircraft, but these qualities change as the regime of flight changes. Because of the V-22's aerodynamic characteristics and its unique cockpit features, a new and more resourceful approach to pilot training must be considered.

In relation to the third issue, the first aviators to be trained to fly the V-22 for the Navy and Marine Corps will be CH-46 and CH-53 pilots. Since students fully prepared in Undergraduate Pilot Training (UPT) to enter Fleet Replacement Squadron (FRS) V-22 training will not be available until 1994, instructor pilots will be trained beginning in 1992 to support the transition training requirements. A four-stage curriculum is envisioned to provide the transition training. The first stage should employ traditional Computer Aided Instruction (CAI) and programmed texts to teach theory and principles of operation. The second stage will likely consist of classroom training complemented by individual practice on ASTs to acquire basic procedural skills. In the third stage, OFTs could be used for training combined crew operations, crew coordination, flying skills, and mission scenarios. During the fourth stage, aircrews would practice in the aircraft. The transition training can be considered a solution for the short-term. In the long-term, students will leave UPT with basic VSTOL flight skills. Mission specific training will be provided in each services' operational training (i.e., FRS) domain. To handle the joint pilot training requirement, an undergraduate consolidation/ co-location is considered a viable option at this time. It is significant to note that the CH-46 and CH-53 pilots selected for V-22 transition training will be leaving cockpits with conventional controls and "steam gauge" displays to assume the roles of highly coordinated system operators in an all-glass cockpit with highly integrated subsystems and fly-by-wire capabilities.

OUTSTANDING TRAINING SYSTEM REQUIREMENTS AND CHALLENGES

In previous sections, we have pointed out how the V-22 differs significantly from other aviation weapon systems and the substantial impacts upon training system design. In this section, we overview two novel commitments important to the V-22 training system design and implementation, the use of the Manned Flight Simulator and the Ada programming language. From there, we direct our attention to requirements and challenges that still must be met to realize maximum training value and effectiveness from the V-22 training system.

The Manned Flight Simulator And Ada

Traditionally in the development of flight simulators, the services have acquired development testing (DT) and operational testing (OT) for trainers from the manufacturer. In the V-22 program, an early commitment was made to utilize the Manned Flight Simulator (MFS) at the Patuxent River, MD Naval Air Test Center (NATC) to support DT and OT as well as the engineering development of the first article trainer. Present plans call for training of 24 pilots from the joint services on the MFS to enable DT and OT to be conducted at NATC. The MFS will feature a modular, strap-down cockpit on a six degree of freedom motion system and will include a wide-angle visual system. The design flexibility incorporated in the MFS will allow its use in future aircraft development programs and will provide for engineering simulation prototyping for future V-22 OFT modifications. A major benefit of the present approach is the corporate, in-house capability the Navy will have at its disposal for bridging engineering simulation and training simulation problem areas in next generation aviation weapon systems.

Another key decision that should have far-ranging implications for the V-22 training system, as well as for the simulation industry at large, was the commitment to use the standard DOD programming language, Ada. The decision to employ Ada for the V-22 trainers should provide a major stimulation to industry to develop and refine capabilities in the Ada application area. The structured design features of Ada should logically complement the services' modular approach to trainer development discussed earlier. Since Ada programs are modular and re-usable by design, configuration management should be greatly assisted and there is promise for major reductions in the total software life cycle costs.

The Requirements And Challenges

At this point, we turn to consider other training issues associated with the uniqueness of the V-22 and the wide variation in mission requirements across the services. Our primary intent is to challenge the training systems community to react creatively in proposing new elements for the V-22 total training system. We must be able to identify the range and domain of the new training requirements very early if the system user is to produce the most highly qualified, combat ready V-22 aviators. Of a long list of problems and associated training needs that can be anticipated, we have selected a few of the more salient upon which to comment.

Systems Management Training Requirements.

Along with the radical changes in aircraft design characteristics have come greater system complexity and the aviator's new role as a systems manager in the V-22. Superior pilots will no longer be the "best sticks" but will be those with the greatest systems knowledge, decision making skills, and ability to maximize the utilization of system resources to achieve mission objectives. The supervisory role of the V-22 aviator in the CMDS environment does not imply, however, that workload will be decreased as a result of the multifunctional displays/controls, the greater information processing performed by the subsystems, or the improved integration of the information in alpha-numeric and pictorial formats. On the contrary, the numerous computer-driven operations performed by the V-22's subsystems present thousands of options or potential ways for pilots to interact with the "layered" information available in the system. Memory and information

processing demands will increase for the V-22 systems manager and in ways that are not easy to predict under the stress of emergency or combat. Moreover, the mental functions and skills required of V-22 aviators in their new systems supervisory roles will not be highly structured activities in most situations. Sequences of specific actions will depend on the events that arise as the operational scenario unfolds. Since the tasks can seldom be described as fixed, deterministic sequences, operators cannot be adequately trained by drill on fixed scenarios. Also, it is difficult to imagine that training on a sufficient number of unplanned conditions could be provided.

In the above discussion, we have alluded to the need for training system capabilities that will ensure delivery of in-depth knowledge and skill bases through extensive exposure to system simulations. Low-cost, microprocessor-based simulations of V-22 displays/controls with scenarios that exercise a very large range of system options, as well as free-play capabilities, are easy means to augment training along these lines. Further discussion of possibilities in this area will follow in a later section. In addition, we have called for a longer term approach that examines the component skills, knowledges, and rules employed by operators in real time to analyze specific conditions and formulate actions to counter threats and achieve mission objectives. Future training for systems managers must focus beyond a procedures orientation upon methods that will enhance an aviator's understanding of total system resources and how these assets can be most effectively utilized.

Crew Coordination Training Requirements.

The combination of the CMDS advanced technology and two systems managers in the V-22 crewstation sets the stage for new kinds of problems. Effective coordination between crew members during mission critical phases will have a major bearing upon operational success. We are concerned now with establishing rules of behavior in dealing with contingencies in high workload, high density threat environments. Technology in the V-22 allows rapid reconfiguration of the entire display/control ensemble as well as helmet-slaved sensor slewing. But creative initiative on the part of crew members in this cockpit could quickly lead to confusion and doubt about system responsibilities or the utility of displayed information. Recent studies of other advanced systems have shown that two crew members perform worse than one under high workload when crew coordination training is lacking. There are numerous examples from military and commercial aviation of the disastrous consequences of breakdowns in crew coordination.

Once again it can be speculated that low-cost simulation technology could be derived to augment OFT and AST capabilities. Enough variety in "canned" sensor imagery and electronic warfare (EW) threat simulation could be achieved at reasonably low cost to make training problems challenging and interesting. The real objective, however, would be to establish the principles and discipline of crew coordination in system operation and resource management. Also, simulation problems with varying workload demands could be designed to challenge crew coordination under stressful conditions.

Mission-Oriented Training Requirements.

The complexity of missions planned for the V-22 implies different training issues than those associated with system complexity. By mission-oriented training we mean training that encompasses intra- and inter-aircraft crew coordination and training in special tactics and missions. Traditionally, fleet training has been designed to maintain proficiency. For the V-22, however, many new and important dimensions of training will begin in the fleet. Moreover, continuous and more highly structured fleet training will be necessary in view of the range of missions and capabilities of the system.

Since a weapon system is seldom launched by itself, there is always a need for integrated air combat tactics training. Other examples of V-22 training conducted by the fleet likely will include formation flying, low level flying, tactical situation assessment, threat avoidance, and forward insertion-extraction procedures. It is the multi-mission capability of the V-22 that may lead to problems in providing adequate training for pilots. Although the V-22 OFTs will comprise numerous advanced capabilities and can support mission-oriented training requirements, the systems could have limited availability under what promises to be a heavy training load. Low-cost, networked simulations with free-play or limited scenario features could be entertained as complements to the OFTs for mission-oriented training. Such systems would need to be flexible, adaptable and contain enough sensor imagery and EW features to ensure continued challenging and novel situations and to provide interest and motivation on the part of pilots. The additional training opportunities provided by the lower cost but sufficiently realistic simulations should provide an expanded base for skill refinement on use of displays/controls and on maximum utilization of system resources to accomplish the mission.

Another technology area that bears watching for trends that will provide new opportunities to enhance the quality of training concerns the integration of mission planning and battle management systems. Once computer-based mission planning becomes a reality, it will relieve the flight crew of most of the tedium of detailed flight planning. The aircrew's time just before a mission could be better spent in verifying the accuracy and completeness of the plans, and in dress rehearsal for the coming mission, the ultimate in realistic training. The drop-in-tape capability of the AYK-14 computer and the commonality of the V-22 with its training simulators should contribute to dress rehearsal capabilities in the future.

Low-Cost Training Simulation Requirements.

Complementing operational practice with simulator time will continue to accelerate, both for economic reasons and because situations can be reproduced and examined that would be unsafe or impractical using actual equipment. From the foregoing views on V-22 training requirements,

the need for additional simulation capabilities to satisfy what appear to be outstanding requirements was frequently referenced. Transition training, UPT and FRS training could substantially benefit from the introduction of well-structured and low-cost computer based training (CBT) that complements the training system design elements already planned. We cannot disregard at this point in time any approach that will better ensure the acquisition, maintenance and enhancement of the broad array of skills that will be required of V-22 aviators.

Lower cost simulations of critical system features must be viewed only as tools for training and not as less than perfect renditions of the operational system. It is also most important to determine the training objective(s) we are trying to achieve through any simulation. Coupled with appropriate curricula, measures of learning and performance proficiency, and useful feedback to students, CBT can potentially enhance numerous dimensions of the training system. Acquisition and life cycle costs for microprocessor hardware are minor compared to the same costs for OFTs. Software development costs for simulation on microprocessors also can be contained by drawing upon software developed in the engineering design process. In some instances, the software developed for engineering simulation can be transferred directly to training applications.

CONCLUSIONS

As innovation has been the keynote in the V-22 weapon system design, innovation must also be applied to the development of the training system. The technically sophisticated V-22 is an excellent example of the joint services' strategy to maintain military superiority in the face of our adversaries' numerical advantages in weapon systems. If the philosophy of superiority through technology is to work, however, then we must field operators and maintainers who can efficiently and effectively get the job done. Therein lies the challenge before the training community - to create the skill bases in personnel necessary to take maximum advantage of our systems' capabilities. The process will admittedly not be simple as we must set aside many time-worn ways of training. The critical advantages that can be realized with the V-22 aircraft will depend largely upon our success in developing a comprehensive and effective training system that yields the most combat ready aviators in the world.

ABOUT THE AUTHORS

Major Schultz received a BS in Chemistry in 1975 and entered the USMC the same year. He was the first fleet CH-53E pilot and the first air officer of the light vehicle battalion. Major Schultz chaired the CH-53D and CH-53E FPT and participated on the CH-46 and AV-8 FPT. He is a graduate of the Defense Systems Management College and currently serves as the program manager for V-22 training systems at the Naval Air Systems Command.

Commander Owens holds a Ph.D. in Experimental Psychology with major area of study in human factors psychology and minor areas in Industrial/Electrical Engineering. He entered the USN Medical Service Corps in 1973 and was designated a Naval Aerospace Experimental Psychologist in 1974. He has served in various assignments related to aviation RDT&E management and execution. Immediately prior to his assignment at the Naval Air Systems Command, he was Head, Psychological Sciences Department, Naval Aerospace Medical Research Laboratory, from 1981-1984. On the staff of the Air Program Coordinator for Training Systems (APC205), Commander Owens is presently serving as Deputy for Plans, Programs, Analysis and Evaluation.

Lieutenant Commander Harris received his commission in the USN Medical Service Corps in 1974. He holds a MS in Psychology and is nearing completion of requirements for a Ph.D. in Industrial Engineering and Operations Research. Prior to his assignment at the Naval Air Test Center (NATC) and out-service training at Virginia Polytechnic Institute, he managed the Navy's voice-interactive systems and technology program at the Naval Air Development Center. Lieutenant Commander Harris developed the first artificial intelligence system, "DEMON", to be flight tested in an F-18, and has received wide recognition for his expertise in the application of advanced technology to crewstation design. His current assignment at NATC is Head, Computer Technology and Simulation Department, Systems Engineering Test Directorate.

THE STRENGTHS, SUCCESSES AND LESSONS LEARNED IN THE USE OF COMPUTER-BASED TRAINING BY THE S-3A, F/A-18 AND F-14A NAVAL AVIATION TRAINING PROGRAMS

Ellen M. Le Vita, M.A.
ManTech Mathetics Corporation
San Diego, CA

Amanda M. Williams, Ph.D.
Tactical and Training Systems Division
Logicon, Inc.
San Diego, CA

ABSTRACT

The role of computer-based training (CBT) is growing in support of high-technology aircrew training systems. As the potential of CBT continues to grow, it is expected to play a more significant role in highly sophisticated training applications. The advantages of CBT are many. It is a medium for both cognitive and procedural training; it is currently the most flexible medium for maintaining concurrency with modern, rapidly changing aircraft and weapons systems; and it can be used as a vehicle to manage instruction. The self-paced capabilities of the medium ensure that students meet criterion levels of performance even when used within the context of lock-step programs.

CBT is being applied in three Navy aircrew training programs. It has been used in the S-3A and F/A-18 programs for several years and is currently being implemented to train F-14A aircrew. Future programs, including the F-14D, A-6F, E-2C and SH-60F, will also use CBT in aircrew training systems.

This paper will describe the strengths, successes and lessons learned in the use of CBT by the S-3A, F/A-18 and F-14A programs and how the use of CBT in these programs can serve as the building blocks for new CBT and training system development. The general conclusions of the authors is that a means to communicate these experiences will allow training systems managers and planners to build programs on a sound basis of experience. In this age of rapid technological advancements, training systems designs based on experience will offer the critical advantage.

INTRODUCTION

The emergence of digitally-based weapons systems has led to a requirement for training systems that are easily and conveniently updated in response to software changes in the aircraft computer systems. Computer-based Training (CBT) is meeting that requirement for both academic and hands-on part task training.

Traditional academic media, such as slide/tapes, workbooks and lectures, require a significant amount of time, money and manpower to keep current with the aircraft. For complex flight simulation including part task training, the cost is even higher. Since the CBT lessons operate from a program source code, a single change to the source updates a lesson. In the case of other academic media, all copies must be individually updated, dramatically increasing the manpower and cost to keep the courseware concurrent. In the case of part task trainers, extensive hardware and software upgrades must be accomplished. A typical CBT lesson can be updated in a matter of minutes, hours or days, while the modifications to other training media and devices takes weeks or months.

During the past few years, the Naval aviation community has attempted to capitalize on the advantages offered by CBT and has incorporated it into their training curricula. The S-3A training program was the first to be developed, followed by the F/A-18 program. The F-14A community is currently introducing CBT into their curriculum. There have been mixed responses to these programs.

PURPOSE OF THIS PAPER

The experiences gained through several years of CBT application in Naval aviation programs can be used to identify management perspectives for future programs. In support of this idea, the purpose of this paper is to briefly summarize some of the problems, successes and lessons learned in its various applications. It shows, by direct examples, how the strengths and weaknesses of one program have provided building blocks for the next program. It also shows how problems solved in one program could have been prevented in others if communications from one project to the next had been formalized.

The use of CBT in Naval aviation has now been extensive enough that a more comprehensive data base can be accumulated. The general conclusions and recommendations of the authors is that this data base should be used to develop a set of guidelines for management of the procurement, development, implementation and maintenance of CBT within aviation training programs. These guidelines should be a "living document" which would be updated at the conclusion of each new development effort. This way, as CBT technology advances, managers would have an up-to-date data base to use in planning decisions.

AIRCRAFT MISSIONS

The aircraft missions of the S-3A, F/A-18 and F-14A are briefly described below. The use of CBT in these programs is described in the next section.

The S-3A "Viking" is a four seat, twin engine, antisubmarine warfare (ASW) sonar aircraft. The primary mission of the S-3A is to locate and identify conventional and nuclear-powered submarines. The Viking carries a comprehensive range of sonobuoys in support of the mission as well as various types of bombs, torpedos and mines. The crew consists of a pilot, copilot, tactical operator and acoustic sensor operator. The training site for the Viking is located at Naval Air Station (NAS) North Island, California.

The F/A-18 "Hornet" is the most recent tactical aircraft to be introduced to the Fleet. The Hornet is a single seat, twin engine, digitially-based weapon system which employs air-to-air missiles, light attack bombs and an internal 20 mm gun in support of its mission. Its mission includes the air-to-air fighter role and the light attack air-to-ground bomber role. The primary training sites are located at NAS Lemoore, California and NAS Cecil Field, Florida.

The F-14A "Tomcat" is a two seat, twin engine, swing-wing aircraft. The Tomcat carries the long range Phoenix missiles, shorter range air-to-air missiles and an internal 20 mm gun. The F-14A missions encompass both the roles of Air Superiority in the defense of the Fleet and the Strike Groups and the lessor role of Tactical Air Reconnaissance Pod System (TARPS) missions. The crew consists of the pilot and radar intercept officer (RIO). The primary training sites are located at NAS Miramar, California and NAS Oceana, Virginia.

CBT IN NAVAL AVIATION TRAINING SYSTEMS

The introduction and implementation of CBT in each aircraft community was accomplished in different manners. In each program there were successes as well as problems.

S-3A

Computer-based training was first introduced into S-3A aircrew training over ten years ago. During the early to mid-seventies, CBT was believed to be the solution to many training problems including the answer to automating training systems. However, due to the novelty of CBT and lack of experience through application, problems arose in many programs and CBT did not meet expectations. This early introduction of CBT into Navy aircrew training represented one of the first applications of CBT in military training. An examination of the experiences gained in this early implementation reveals the bases of some of the general problems encountered in many programs.

The initial CBT courseware for S-3A training was developed under two successive contracting efforts and was retrofit into an already existing training program. The concept of using CBT for simulation exercises and part task training had not yet been introduced, thus the first lessons were academic in nature. The lessons were interactive in the sense that after the student read through textual material, he was required to answer questions on the material by selecting from alternatives. Further courseware development under a second contracting effort included the use of CBT for simulation exercises to download some training from an aircrew position trainer.

One of the first problems with courseware development during these early efforts arose when the second effort had to be performed off-site. Although the first contractor had worked on-site at the training squadron, due to a problem with available on-site computer resources, the second effort was performed off-site at the contractor's plant. This immediately posed problems with respect to the need of a close liaison with the training squadron.

The second contractor worked off-site, delivering lessons as they were developed to the training squadron. While this system worked to some degree, the contractor ended up spending much time traveling to the site and would have preferred to be resident during the courseware development.

During the development, problems with reviewing the Lesson Specifications and CBT lessons by squadron subject matter experts (SMEs) arose. The SMEs did not know how to review these lesson materials. Many of the changes the SMEs requested were due to differences in their instructional styles rather than technical problems. In addition, because Navy aviation officers have two jobs, flying and their squadron duties, many were too busy to review the lessons in a timely manner. Thus, the turnaround on the lessons was very long, in some cases causing the contractor difficulty in meeting milestones and delivery dates.

To solve these problems, the contractor took two actions. First, a SME training course was offered. This was a one day course offered at the training squadron. All SMEs reviewing lessons were required to take the course. Second, their reviews had to culminate in one of three results: a) lesson accepted as is; b) lesson accepted as changed; or c) lesson ignored. Lesson Specifications or CBT lessons which were ignored were considered accepted as is after 30 days. This approach encouraged SMEs to look at the lessons in a more timely manner and if they were acceptable no further action was required of them thus saving busy aviators valuable time.

Another problem arose with respect to an older courseware development language used for the first set of lessons. An update to the CBT operating system required many of these lessons to be recompiled to run. Although civil service programmers were in place to operate the CBT system, the definition of their job roles did not include maintaining and updating the courseware. As a result, many of the early lessons were no longer operable. In addition, changes to the aircraft and/or learning syllabus resulted in the need to update courseware. Over time, as the CBT courseware became outdated, the CBT system was utilized less often.

The initial CBT courseware was not well accepted by the squadron training personnel or replacement aircrew. This was probably due to a combination of factors. CBT was a new, untried instructional medium in aircrew training; there were problems of keeping CBT concurrent with the aircraft; CBT was retrofit into an already existing training curriculum. At some point during the program, alternate media such as workbooks were introduced

that allowed the students to obtain the same materials found in the CBT lessons. In addition, a very long and detailed CBT lesson was developed and used to describe to students how to interact with the computer system. All this combined to lower the acceptance of the CBT system.

Although this early implementation of CBT was somewhat problematic, the experiences gained and lessons learned helped build toward success in other programs and in a later reconfiguration of this one. Lessons learned included:

- o Subject matter experts must be trained to properly review written lesson materials and on-line lessons.
- o Timely review by busy subject matter experts is difficult to obtain. Procedures must be established to ensure reviews that meet courseware delivery schedules, but that require minimal intrusion into operational schedules. This problem has surfaced in other programs as well.
- o Due to the very high turnover of subject matter experts, an agreement must be reached that prior approval by predecessors is not arbitrarily changed due to personal preference or factors other than technical accuracy.
- o To ensure close communication between the squadron and the courseware developer, lessons whould be developed on-site whenever possible.
- o Tasking must be assigned for the maintenance and update of lessons once they are developed. This tasking should include a liaison function with the squadron training department to ensure that weapons systems, tactics and training objectives modifications are anticipated by CBT staff whenever possible.
- o A means to keep the CBT concurrent with the aircraft would eliminate the need for alternate media. Concise introductions to the use of CBT would result in CBT being more acceptable to the students and instructors.

F/A-18

The F/A-1B pilot training program commenced in August 19B2. A detailed description of the program including the use of CBT is provided in Williams (1). The potential for extended applications to tactical training scenarios is discussed in Williams-Easter (2).

The F/A-1B CBT development benefited and capitalized on the experience gained in the S-3A community. As a result, this program encountered far fewer problems. The development of the entire F/A-1B training system adhered closely to MIL-T-29053B, the standard provided by the Navy for Instructional Systems Development (ISD) efforts. This standard provides guidelines for the development and implementation of all facets of a training system.

Ongoing modification and management of all of the training media, including CBT, was planned and provided for by a management structure in which the ISD officer was responsible for all lesson materials. The management of the F/A-1B training program was described in detail in Rondestvedt (3). The ISD Department, as a whole, includes squadron military personnel, government service training specialists and on-site contractor support. The ISD officer coordinates all activities in support of the training system. When potential deficiencies are identified, the Training Department determines whether a change is required and forwards a formal request to the ISD Department. Subsequently, ISD implements the lesson change, which goes back to the Training Department for final approval.

Initially, the courseware was developed at the contractor's plant. Subject matter reviews of the courseware was accomplished by squadron personnel traveling to the contractor's plant. Communication problems soon arose. The Navy requested the presence of on-site CBT development personnel to fine tune the lessons as they arrived from the plant. Presently, an on-site contractor performs all lesson development and updates.

Each of the CBT lessons in the F/A-1B program contains interactive portions and the interest level of the students in the lessons and their acceptance of CBT is very high (2). Each lesson is composed of several segments corresponding to one or two training objectives. Each segment presents a text to introduce and discuss the objective(s) and proceeds to an interactive section that requires actions on the part of the students. In addition to holding students' interest, the part task trainer application of CBT allows students to practice procedures that would formerly have been trained and practiced in either a flight simulator or the aircraft. Therefore, the use of CBT as a part task trainer reserves these more valuable and expensive resources for the training of more complicated skills. Transfer-of-training from the CBT procedural lessons to the aircraft is evidenced by the successful demonstration of these skills in subsequent simulator or aircraft training sorties.

As part of a goal to provide maximum, easy accessibility to the F/A-1B courseware, the students' introduction to the CBT system consists of a short oral briefing to an entire class. Students receive the briefing while seated at the CBT terminals and are stepped through procedures to sign on, sign off, locate lessons, etc. After all students know the basics, they are stepped through the first segment of the first lesson. Since all the segments of the CBT lessons are structured in the same way, after the completion of the first segment students know how to use the CBT system. The process of introducing an entire class to the CBT system and training them to use it takes less than an hour.

Finally, a CBT Training Specialist, whose first task was to guide the implementation of CBT within the F/A-1B program, was assigned. In addition to directing all activities in the F/A-1B self-paced learning center, the Training Specialist also has the responsibility for coordinating all development and maintenance efforts of the contractor, the F/A-1B Training Squadron and the training device support detachment personnel. All procedures for student use, grade reporting, courseware standards and CBT system operation are established by the CBT Training Specialist.

In summary, the use of CBT in the F/A-18 training program was seen as extremely successful. However, this was to a great extent due to the experiences gained in the S-3A program. Most importantly, some of problems that had surfaced in the former program were able to be anticipated and avoided and, therefore, facilitated a smoother integration of CBT into the F/A-18 training system.

Some of the more important considerations in the implementation of the F/A-18 CBT system included:

- o The CBT system, along with the entire training program, was implemented and then conducted according to an established guideline. In addition, this guideline continued to be closely followed. This provided structure for the program as a whole and helped to ensure the smooth integration of all of the individual media and training devices, including CBT.

- o Assets required to keep the courseware concurrent with the aircraft and squadron training objectives are controlled by the ISD officer. The ISD Department makes modifications at the formal request of the Training Department.

- o The presence of on-site lesson support ensures the concurrency of the CBT courseware with the aircraft and weapons systems, the compatibility of the courseware with releases of the CBT operating system and conformity with training squadron objectives. This helped to avoid the communication problems between the contractor and the squadron that tended to exist in the S-3A program.

- o A short briefing was used to introduce a class to the CBT system. This concise, convenient introduction makes the CBT system more accessible to students. This is in contrast to longer, on-line introductory lesson which had been used in the S-3A program.

- o A CBT Training Specialist was assigned to manage the implementation and use of CBT. This position provided a focal point for the use and management of the CBT system within the training program.

S-3A Update

In response to an identified need in the S-3A program to download training from a heavily used position simulator, an ambitious program of new courseware development was undertaken.

In this new implementation, the CBT lessons were designed exclusively as part task training segments for procedures to operate equipment in the S-3A aircraft. In contrast to earlier development in the S-3A and F/A-18 program, this large effort was conducted solely by government personnel; no contractors were involved. As in the F/A-18 program, a Training Specialist took a strong hand in coordinating and managing the effort.

In summary, as a result of both the initial implementation of CBT in the S-3A program and the F/A-18 program, this revision of the CBT system proved to be a very successful effort. Specific building blocks included:

- o The demonstrated success of the previous use of CBT as a part task trainer led to the increased use of this application in the S-3A program.

- o As in the F/A-18 program, a single focal point provided by the Training Specialist helped ensure success of the program.

F-14A

The F-14A CBT development is currently in the implementation phase of the initial development process. The ongoing status of this project offers the opportunity for a more detailed look at the development process. The F-14A CBT development, like the S-3A, is being retrofit into an already existing training curriculum. Thirty-five lessons were developed by the contractor using a team of contractor Subject Matter Experts (SMEs), Instructional Psychologists, Artists and a Programmer.

The F-14A Training Squadron based their CBT development on a different philosophy than did the F/A-18 Training Squadron. The F-14A Training Squadron chose to complement the existing training system by using CBT purely as a procedural part task trainer. The emphasis on the use of CBT in the F-14A program for simulation and part task training, capitalizes on the interactive capabilities of CBT. The CBT was viewed as an intermediate step before the students went to the simulators and aircraft. The F-14A Training Squadron expected students to obtain the basic aircraft systems information from the Naval Air Training and Operating Procedures Standardization (NATOPS) and lectures. In contrast, in the F/A-18 training system, the CBT covered the basic information as well as providing some simulation. The use of the NATOPS as an instructional medium was was not part of the formal syllabus although it remained the primary technical and operating procedures manual. For details on the CBT design used in the F-14A program, see Randel (4).

The F-14A development, like the F/A-18, followed the guidelines and standards established in MIL-T-29053B. However, a problem arose over what a Lesson Specification should constitute. The contractors delivered a standard specification but the Training Squadron was expecting something more along the lines of a storyboard. A compromise was established wherein the Lesson Specifications included many aspects normally found in the storyboard stage.

Another problem that arose early in the development process was that the Navy SMEs did not fully understand their responsibilities when reviewing preliminary materials for the CBT. The Training Officer, who served as the liaison between the SMEs and the contractor, would pass the Lesson Specifications along to the SME without specific instructions or guidelines on how to review the materials although these had been made available by the contractor. As a result, the Lesson Specifications would be reviewed for technical accuracy without thought as to how it would transfer to the interactive medium of CBT.

Lesson Specifications approved by the SMEs were subsequently storyboarded and programmed. When the SMEs would review the same lessons on-line, they requested changes to material they had previously approved. These changes often resulted in duplications of effort and could have been identified at the Lesson Specification phase.

An additional problem resulted from writing all the Lesson Specifications prior to any of the actual CBT development. Too much time elapsed between the start of an individual Lesson Specification and its draft acceptance on-line. Due to collateral duties, Navy reviews ranged from two weeks to over six months. Authors had to reread the NATOPS and relearn the details in order to write the storyboards.

Because of the extended length of time for complete development of each lesson, including the review process, there was little continuity of Navy SMEs. The SME that reviewed and accepted the initial draft Lesson Specification was seldom the SME that accepted the final CBT lesson, or in a number of cases, even the draft CBT lesson. In some extreme cases, each Navy review was conducted by a different SME, each requiring changes in style of lesson presentation rather than changes due to technical inaccuracies.

Finally, as is the case with all aircraft and weapon systems, modifications are made periodically. During the F-14A CBT development process, two modifications to the weapon system program tape were released. The second change occurred after many of the lessons had been accepted in draft form but prior to final acceptance. Therefore, many of the draft lessons that had been approved pending minor revisions suddenly required major rewrites prior to final acceptance. In addition, new lessons were still being storyboarded and programmed but because the other revisions also had to be made during the same time frame, less hours were available to devote to new lessonware development. Computer-based Training lends itself well to these types of updates and changes. However, the integration of such modifications should be carefully planned, when possible, to ensure it does not interfere with ongoing lesson development.

The courseware began being implemented in April, 1987. The first four classes were allowed to review the courseware on an optional basis. That is, slide/tapes continued to be required training materials while the CBT was on an "as time permits" basis. If the students desired to view the CBT, they could arrange to do so during their extra time.

During the implementation phase, only three computer stations were available for student use. The squadron has identified the need to buy a minimum of nine additional student stations and have the available funds. However, no contract vehicle currently exists through which the squadron can buy the hardware. As a result, although many of the students have used the CBT system diligently, others have not been able to use it because there were no terminals available during their free time.

One last point should be noted. Although not perceived as a problem at the outset of the F-14A development phase nor at the time of the S-3A development, the issue of retrofitting a new training medium into an already existing program requires special attention. The two CBT programs received more resistance and had more logistical problems than did the F/A-18 CBT program which was developed as a intregal part of the entire F/A-18 training curriculum. The contrast between the programs was especially apparent because of the overwhelming acceptance by all parties concerned in the F/A-18 program. The S-3A and F-14A CBT lessons appeared to be readily accepted by the students but the instructors were more reluctant about its instructional value.

In summary, the F-14A development built on previous development in the S-3A and F/A-18 programs.

- As in the revised S-3A program, the emphasis was to use the CBT system for part task training of procedures in a totally interactive mode.
- Using the philosophy of the F/A-18 program, a short oral briefing, followed by a brief on-line lesson is used to introduce students to CBT. Total orientation time is under one-half hour.
- From the problems encountered in the S-3A and initially in the F/A-18 programs, it was seen as mandatory to place the CBT development team on-site.
- Although problems arose (as discussed below), as in the F/A-18 program, the military specifications provided standard guidelines for lesson development and courseware management.

Because the F-14A development effort is in progress at the writing of this article, several problems are currently being addressed. These have been discussed in detail above and include:

- Contractors and squadron personnel had different expectations of what the deliverables required. A Military Specification (MIL-SPEC) outlining the format of CBT-associated deliverables should be developed.
- SMEs accepted Lesson Specifications without having sufficient knowledge of CBT. The contractor should have more direct access to SMEs to instruct them on review procedures, provide guidelines for acceptance of materials and to place emphasis on the importance of the paper reviews prior to the on-line reviews.
- Too much time elasped between the initiation of the Lesson Specifications and the completion of draft lessons. Timely review periods should be scheduled and the schedules should be observed. This would help to ensure continuity of SMEs reviewing the deliverables.
- Student reactions, as well as most instructor reactions, to the F-14A CBT has been extremely positive. However, in a few cases negative reactions were encountered. This was perhaps due to these instructors being satisfied with the existing training program and reluctant to change it. CBT should be part of the complete training system when new aircraft are introduced into

the Fleet such as the F-14D or when major modifications to existing aircraft are made, as is the case with the F-14A+. If CBT is to be retrofit into existing training, there should be careful consideration and planning so as to ensure the CBT is well integrated into the training syllabus at the time of implementation.

- The hardware requirements were not met by implementation of the courseware into the syllabus. The money and contracting vehicle for all pertinent purchases required for CBT development and implementation should be identified and ensured prior to the start of any CBT development effort.

CONCLUSIONS AND RECOMMENDATIONS

The Navy has implemented CBT in three different aviation training programs. In each case different conditions existed. In the initial S-3A program, on-site and off-site contractors developed courseware that was retrofit into an already existing program. In the F/A-18 program, CBT was acquired as part of the entire training system procurement. In the revised S-3A program, the courseware was developed entirely by government personnel. Finally, in the current F-14A effort, an on-site contractor is developing courseware to be retrofit into the training curriculum.

Despite the different conditions in which CBT has been implemented, it has been shown that lessons learned in one program provide the building blocks for successive programs. With the impending procurement of several new aviation systems, it would be most advantageous to develop a general set of guidelines which specifically address issues surrounding Computer-based Training in such programs. This set of guidelines should be kept up-to-date so that each development effort can benefit from previous lessons learned.

While some of the lessons learned in the CBT development efforts were communicated and served as building blocks, others were not. For example, the problems of SME reviews pervaded all three programs. In each case, the problem was managed or resolved. A communication vehicle which formally documents such problems and how they were resolved may have helped to avoid these problems in subsequent programs.

The authors suggest that the general set of guidelines being proposed be established as a "living document." This document would be updated by each contractor during or at the conclusion of each CBT development program. These updates or reports on lessons learned would be an identified task in the Statement of Work for every CBT contract. This document, including general guidelines and lessons learned, would be available both to those under contract and, perhaps more importantly, to those preparing proposals. The availability of such a document at the time of proposal preparation would help to ensure much more accurate cost and planning estimates.

A management perspective that looks ahead and anticipates problems must be developed. Prior to the implementation of the three CBT programs discussed in this paper, this would not have been possible. It would have been impossible to anticipate the various types of problems that occurred. However, the accumulated experience with CBT could provide a comprehensive data base to support the development of a set of guidelines for procurement, development, implementation and maintenance of Computer-based Training.

REFERENCES

(1) Williams, A. M. The application of computer-assisted instruction in F/A-18 pilot training. Training Technology Journal, 1984, 1(4), 40-47.

(2) Williams-Easter, A. M. Computer-based training to operate state-of-the-art weapons systems. Proceedings of the Interservice/Industry Training Systems Conference, Volume I. Salt Lake City, Utah, November 1986, 154-159.

(3) Rondestvedt, C. R. The F/A-18 instructional system development program: A military user management viewpoint. Proceedings of the Interservice/Industry Training Equipment Conference, Volume I. Washington, D. C., October 1984, 409-416.

(4) Randel, J. R. Two for the price of one: Procedural tutorial and testing in aircrew training. Paper presented at the annual meeting of the Military Testing Association, Mystic, Connecticut, October 1986.

ABOUT THE AUTHORS

ELLEN M. LE VITA is a Senior Instructional Psychologist with ManTech Mathetics Corporation. For the past two years, she has been developing the F-14A Computer-based Training performing all aspects of the CBT development and management, including designing, storyboarding, programming and computer system operations. Prior to the F-14A project, Ms. Le Vita worked on various aviation and ASW projects primarily developing training materials using the Instructional Systems Development principles. She has also performed a Problem Analysis of the F/A-18 Fuel System Maintenance Training program and designed various experiments to help determine the most cost effective visual system for the T-45. She has published research in the areas of memory and developmental psychophysiology. Ms. Le Vita holds an MA in Experimental Psychology from San Diego State University.

DR. AMANDA M. WILLIAMS is a member of the Technical Staff with the Tactical and Training Systems Division of Logicon, Inc., where she has been involved in Instructor/Operation Station (IOS) design, training systems analyses and Air Combat Maneuvering research. She was formerly the Computer-based Training Specialist for the F/A-18 Pilot Training Program where she was responsible for the F/A-18 Self-Paced Learning Center. Dr. Williams has published in the areas of visual perception, applied visual perception, IOS design, performance measurement and computer-based training. She holds a Ph.D. in Cognitive Psychology from the State University of New York.

WILL PMS MEET THE NEEDS OF A UNIVERSAL AUTHORING SYSTEM?

James R. Stonge
Senior Instructional Systems Engineer
Integrated Logistics Systems
Aircraft Systems Division of Grumman
Linda G. Jensen
Instructional Design Engineer
Display/Trainer Products
Electronics Systems Division of Grumman
Bethpage, N.Y. 11714-3585

ABSTRACT

At the recent Air Force Technology in Training and Education (TITE) Conference in March, a call was made for a universal authoring system. This call was based on the recent proliferation in hardware and applications software technologies during major acquisitions for training by all of the military services. The TITE presentation by Linda Jensen identified the inconsistencies between various hardware components, applications software packages, and actual training applications as a major drawback to effective and efficient government training programs. Part of the problem has been the cornucopia of "authoring systems" that have been created, marketed, and sold. These authoring systems have had three major drawbacks: they are usually incompatible with other authoring systems, they usually have limited applicability, working only with certain specific hardware and software systems, and, more importantly, they are not truly authoring systems. They are programming systems that assume the materials have already been authored and created. The paper presented at TITE addressed the need for an authoring system that works at all stages of a project, identified the general characteristics required of such an authoring package, and specified the capabilities and mechanisms that must be included. Further, it called for a standard that allows compatibility from one delivery system to another, regardless of hardware and applications software. The U.S. Army has identified a standard that approaches this ideal authoring system, the Production Management System (PMS). This paper compares PMS to the ideal authoring system described in the TITE paper and presentation. All of the requisite characteristics comprising a complete instructional system development tool are summarized, including electronic storyboarding, data management, media production, and lesson programming. The paper then looks at each of the characteristics and identifies the ability of PMS to meet the need. It also identifies any PMS capabilities not addressed in the characteristics of the universal authoring system, and evaluates those characteristics for inclusion in the proposed standard. The paper concludes with an analysis of the overall capability of PMS to serve as a universal authoring system, and specifies what capabilities need to be added to make it fully functional.

INTRODUCTION AND OVERVIEW

Users of interactive media continue to face problems of incompatible hardware, software and authoring systems. While the problem is shared by all users, the greatest impact is on the Department of Defense due to the proliferation of delivery systems and software developed over time for a variety of projects. Materials developed under one contract cannot be used with equipment and software acquired via a second contract. Of all media, only interactive media lack universal capability for distribution and transfer across users. This incompatibility is an outgrowth of the lack of comprehensive industry standards. Hardware manufacturers and software developers attempt to create new and unique packages, often with the deliberate intent of preventing users from recycling materials developed for one system for use on another. This strategy has been used not only between rival companies, but by major industry leaders when introducing a new product line that is completely different from previously marketed systems.

The only way to alleviate this situation is to create industry standards such as were instituted for such diverse products as videotapes, tires and light bulbs. When standards are in effect, competition still thrives as producers identify new and better products within the framework of the standards. Thus, the consumer does not need to buy new lamps every time a manufacturer introduces a new light bulb. The industry needs a set of standards that can evolve, but that will allow the consumer (e.g. the Department of defense) to continue to use and update existing lesson materials, regardless of changes to hardware and operating software. In addition, new lessons could be run on 'old' equipment and software regardless of the particular authoring package used to create them.

Two recent events indicate a desire to establish those standards: The US Army identified a standard hardware and software system - the hardware is EIDS, the software is PMS. And, at the recent Air Force Technology in Training and and Education (TITE) conference, Linda Jensen identified and proposed a candidate universal authoring capability. In the case of EIDS and PMS, the specification and deliveries of hardware and software are real. The universal system identified is an ideal that has not yet been realized.

This paper addresses a comparison of PMS to this published ideal system. It specifically avoids a comparison of PMS to any other existing authoring system as no other authoring system has been officially identified as a standard. It also avoids making a value judgement, as it is recognized that PMS was not developed in accordance with the ideal system's specific characteristics and capabilities. Rather, this paper is limited to an evaluation of the relationship of PMS to the ideal, and to the steps required to enhance PMS to meet the needs of the users of the STANDARD, or ideal, system.

THE UNIVERSAL AUTHORING SYSTEM

The ideal system must be a universal authoring system. Three definitions are basic to understanding the characteristics of this system. First, interactive authoring must be defined as a system that allows stimulus and response for both the student/user and the interactive media. Secondly, interactive authoring must provide for both the management of instruction and the record keeping of student performance. Finally,

interactive authoring should be defined as the process of creating the total environment of the delivery system. These three definitions address the practical and philosophical differences evident in the viewpoints of student, instructor, and producer.

PRODUCTION MANAGEMENT SYSTEM

The US Army, in conjunction with Computer Science Corporation (CSC), developed the Production Management System (PMS) software package to use with interim EIDS (Electronic Information Delivery System). PMS is the only authorized software system for interim EIDS. Further, it is being converted for use with the production EIDS, and will be renamed "EIDS-ASSIST." For the purposes of this discussion, PMS will refer to both PMS and EIDS-ASSIST. The version of PMS used for the analysis is version 3.11, because it was the latest version available, and Army sources state that it is identical in capability to the first version of EIDS-ASSIST.

HARDWARE

It has become a popular attribute for successful commercial software applications to be usable with a variety of computers and peripherals. Sometimes this is done by "installing" the software in which the particular equipment is identified during initiation. In other cases the software is sold for particular configurations. In many cases the product of these packages is transferable from one system to another, via modem, disk transfer, or other media. This same concept should be applicable to authoring.

IDEAL

The ideal authoring system must be able to operate regardless of the hardware configuration of either the development station or the delivery station. This is critical if the DoD is to obtain maximum benefit for existing and future resources. Therefore, the ideal will be able to meet the following requirements:

- o Use any brand of videodisc player
- o Use any and all videodisc related peripherals
  - Compressed audio encoders/decoders
  - Compact disks (CD ROM or CD/I)
- o Accept any student input devices:
  - Light pen
  - Touch sensitive screen
  - Keyboard
  - Special function keypad
  - Joystick
  - Mouse
  - Trackball
  - Voice recognition
  - 3d simulator
- o Mount on any computer
- o Accept and use any computer related peripherals:
  - Local Area Network (LAN) interfaces
  - Computer graphics and text generation
  - Mass memory storage
  - Printer interfaces

PMS

PMS is specifically designed to work with the US Army's Electronic Information Deliver System (EIDS), and is not intended to be used with any other hardware. EIDS is basically a PC DOS based computer with limited graphics capability and interface with a Pioneer videodisc play. It is intended to ultimately include the capability for a variety of peripherals, but the anticipated first production capability will have only a light pen interface for students. Production EIDS was not available at the time this paper was written, but is anticipated to be available by late fall/early winter of 1987. The interim-EIDS is a SONY based machine that is generally unavailable for purchase.

AUTOMATED STORYBOARDING

Authoring in any other field of endeavor encompasses all stages of a project. To "author" means to create the total package, whether it be a book, software program, or slide/tape show. Therefore, authoring will include the initial stages of lesson development. For the purposes of this discussion, it will be assumed that all analyses and planning have previously been accomplished. While it is acknowledged that the ideal system will ultimately incorporate the analysis stages, and probably include some intelligence to assist the developers, it is premature to include it in this discussion.

IDEAL

In the ideal authoring system, flowcharting can be accomplished prior to storyboarding, in conjunction with storyboarding, or be an output of the storyboarding process. The ideal system will support any of these choices. In addition, the ideal system will have the following capabilities:

- o Generation of storyboard content:
  - Store all relevant information for later retrieval and manipulation
  - Allow flexibility for author to identify as little or as much information and documentation as is required
  - Use English language to the maximum extent; minimize abbreviations and coding
  - Allow creation of crude (stick figure) drawings as part of the documentation, both on-screen and hard-copy printout
- o Cull, sort and print:
  - The narrative script for narration production
  - All artwork requirements
  - All shot sheets, motion sequences and still frames
  - All video text requirements
  - All special effects
- o Support capabilities:
  - Immediate on-screen viewing of computer generated art or text, during development stage
  - On-screen help available at the developer's request
  - Copy, delete, add or modify format, content and sequence of instruction at any time without paging through menus
- o Lesson programming, to be discussed later, should be fully supported during storyboarding

PMS

PMS uses the utility called PMS I for electronic storyboarding. Much, but not all, of the information entered during this process will

be carried forward during other stages of the total PMS process. This utility allows identification of specific elements of the lesson. Not all information can be identified and recorded at this time -- for example, amplification of programming requirements and full pages of test. Specifically, the developer can identify:

- o The RECORD ID, or event number, which is the unique identifier for that information throughout the total process
- o A description of the general scene content (SG), which is inended for video production
- o A description of the specific scene content (SS), also used in video production, and used as sorting criteria for printing of the production schedule when preliminary numerical codes are inserted prior to the description and specific content abbreviations are used during the description
- o The SOURCE Reference, usually technical manual references, coded to allow review of technical source material
- o Programming functions, (PF1 thru PF4), which are described in detail in that paragraph, and at this point are limited to coded abbreviations of the type of function, without details
- o RESPONSES, which are the branching directions
- o PROCEDURE NAME and PROCEDURE TYPE, identifiers used by the developer, to code groupings of instruction (NOTE: All tests must include the word TEST in the PROC NAME and use the code T in PROC CODE)
- o Three fields of comments (C1 thru C3), which can be used for almost anything, but include the following specific characteristics
  - Only the C3 filed is carried through to the programming phase (PMS III), so programming comments are restricted to this field
  - Sorting (and printing) can be accomplished inserting one of several specific codes as the first entry in the comment field
- o Four narration fields (N1 thru N4), which have uses other than narration, have the following characteristics:
  - Screen text and audio are both identified in the narration fields
  - Space is limited, so comments may have to be carried onto additional records
  - Specific codes are used to identify, for later sorting and implementation, the exact kind of "narration" intended
  - Only motion based audio is acceptable
  - Only two lines of computer generated text can be overlayed on video, each thirty six characters in length, at this stage of the process

In addition, PMS I is used to print the production schedule, as will be discussed in the paragraph on production and post-production.

VIDEO PRODUCTION AND POST-PRODUCTION

Most "authoring" systems do not have any application during the video production and post-production stages of a project, because they require the videodisc to be complete prior to "authoring." As with storyboarding, the video applications are an integral part of the interactive product, and must be considered as part of the authoring process.

IDEAL

As identified in the capabilities for automated storyboarding, the ideal system will sort and print all production requirements, in the format required by the producers. Further, the ideal system will provide the capability to update the data base during the production and post-production phases. This task could be simplified if the ideal system also created a production schedule, including production sequences. It must also be capable of working when planned sequences or schedules have been destroyed by circumstances. In addition, the ideal system will:

- o Permit documentation of source locations for completed production, regardless of media:
  - Electronic, automatic updates of the data base by interface with production equipment whenever possible
  - Manual update of the database regardless of electronic interfaces
  - Documentation of multiple sources and/or multiple locations
- o Automatic creation, storage and documentation of all video text without additional human intervention
- o Permit retrieval of all source material for rough editing:
  - Independent viewing of all source materials
  - Identification of selected source materials for final editing (post-production)
  - Call up of multiple sources, and manipulation of those sources, with automatic updating to database
- o Create an edit decision list (EDL) based on selections made during rough editing:
  - All decisions will be retained, including special effects
  - Review (or preview) of edit possible at any time
  - Manual additions, deletions, or modifications to the EDL
- o Provide complete control of the final editing (post-production) process:
  - Computer control of the total editing process
  - Single keystroke implementation of the EDL
  - Automatic update to the database via electronic interface, include SMPTE numbers and videodisc frame numbers

PMS

Video production and post-production are accomplished primarily using the software routines embodied by VDB I, PMS II, and VDB II. When the original storyboard laying out the video requirements is complete, a conversion process is used to create the first stage video data base (VDB I). This feature is arguably the friendliest and most powerful tool in the PMS repetoire. It is intended to be used in conjunction with the PMS I paper Production Schedule as an electronic shot sheet in situations where the computer can be set up at the production location and data entered during the shoot. The VBD I features include:

- o Reordering VDB I data files
- o Display/Edit files, when files are ordered alphabetically by subject (SG-SS)
  - Beginning with first record

- Beginning with a specific Record ID
- Skip over Records not yet shot
- Skip over records added to the data base
- New records may be added for reference, but will not be automatically carried into later processes
- o Record Shot numbers
  - Beginning with first unrecorded scene
  - Begin at specified Record ID
  - Continue from previous session, if any
- o Post SMPTE codes
- o Report status at any time
  - Find all skipped records
  - Print the file in one of two formats
    - a. Alphabetical, by subject (SG-SS)
    - b. Numerical, by shot number

With the VDB I inputs complete, another conversion process posts the recorded SMPTE codes to the PMS II data base. PMS II is then used to update the storyboard and produce the edit decision list (EDL). Updating the storyboard includes the following:

- o Verify that the narration and captions from PMS I are still appropriate to the actual video material obtained
  - Correct and/or add SMPTE codes
  - Enter any desired character generator storage identification codes
- o Manually incorporate any additional records identified in VDB I that did not come from PMS I

Once all additions and deletions have been completed, the Edit Decision List is assembled and printed. This report is a detailed listing of the VDB II data to be used to manage post-production, much as the Production Schedule is a blue-print for production. It can be constructed in different ways to support various project requirements:

- o Print all records in record ID order
- o Print only Motion Sequences in Record ID order
- o Print of still frames in SMPTE Code Order

The EDL(S) are then used in conjunction with VDB II to manage post-production. VDB II is an electronic EDL with two modes of operations, one for relatively small, linear or less complex projects under 1000 records or events, and one for large, complex projects. Mode I uses these capabilities:

- o Record Premaster SMPTE codes in one of two formats
  - Record ID order
  - Composed SMPTE order
- o Display/Edit VDB II files in one of four numerical orders
  - Production SMPTE
  - Composed Identifier
  - Premaster SMPTE
  - Record ID
- o Remove deleted VDB II records
- o Post premaster SMPTE and frame number to PMS III data file

Mode II adds these additional features:

- o Record composed edits
- o Print the file

LESSON PROGRAMMING

Experience has shown that if the developer is not involved in the lesson programming, deviations from the developer's intentions are possible, if not likely. Further, errors in transcription are common. In addition, as there is usually some period of time between writing of the lesson and the programming, some of the original intent may be forgotten even by the developer unless detailed and time-consuming documentation is prepared. Experience has also shown that if programming requirements are not considered at the time of lesson development, the lesson may be incomplete or, even worse, impossible to implement as designed. This can all be solved if programming is accomplished simultaneously with storyboarding. Therefore, the ability of the developer, who may not have computer programming skills, to program while storyboarding is essential.

IDEAL

The following characteristics are associated with the ideal system's programming capabilities:

- o All programming will be via English language commands:
  - Shortened forms are acceptable
  - Minimal amplification will be required
- o Complete flexibility will be provided:
  - All concurrent and/or sequential system performance prior to a student input can be accomplished in a single event or record
  - Sequencing of system performance can be in any order, at the discretion of the developer
  - No artificial limitations such as the number of branches, number of computer generated text or graphic overlays, and other system capabilities, will be imposed on the developer
  - Format, structure, sequence, and/or content can be copied, added, deleted, and/or modified at any time
  - New system capabilities can be added at any time, with immediate inclusion by the developer, even if the hardware and/or software for the features has not been finalized
- o The developer will have lesson control of:
  - All video commands (assuming player has the capability), including forward, reverse, single frame, variable slow motion, and animation (rapid toggle between two adjacent frames)
  - All audio commands, including disc track 1, track 2, both tracks, no tracks, retrieval and playback of digitally stored audio (on videodisc or computer disk), and audio stored on secondary media such as tape or compact disk
  - computer generated text or graphics, including the ability to generate in background mode for instant display when needed
  - Interface with student input media, including simulator(s) when appropriate, for both stimulus and response
  - All CMI functions
- o Computer Management of Instruction (CMI) capabilities:
  - Identification of response as right or wrong, including, as a minimum, three levels of wrong responses; critical or fatal error, major error, and minor error
  - Branching within a lesson based on student performance
  - Branching between lessons based on

- student performance
- Random selection of a small number of problems from a large pool, and random ordering of a fixed set of test questions

PMS

While elements of lesson programming are completed in earlier phases, the complete and final lesson programming is accomplished using PMS III, after the videodisc has been manufactured. The following paragraphs identify the various programming functions, according to order of use:

PMS I. During PMS I, as described in the paragraph on Automated Storyboarding, the following programming functions are accomplished:

- o Identification of RECORD ID, or event number
- o Identification of branching based on RESPONSE
- o Specification of computer text, up to two lines of 36 characters each
- o Preliminary identification of programming functions
- o Beginning and end of tests (BEG END)
- o Comments related to programming, in comment field C3

VDB II. PMS II provides no additional programming capabilities, but is the basis for VDB II. In this video data base file, the primary functions are related to post-production, but the entry of SMPTE numbers allows the system to automatically create the videodisc frame numbers required as part of the programming activity. This conversion of EDL SMPTE numbers to videodisc frame numbers, inserted into the RECORD ID's, is a programming feature that enhances PMS.

PMS III. In this last of the PMS stages, the majority of the PMS programming is accomplished. While programming related functions continue, such as testing and debugging, all final additions, deletions, and modifications are accomplished using PMS III. All of the programming is accomplished by amplifying previously entered data. Specific programming functions accomplished in this stage include:

- o Identification of audio tracks for motion sequences (MO)
- o Specification of the length of time, in seconds, a time still (TS) will appear on the screen
- o Specification of screen location from computer graphics
- o Specification of color and location of computer generated text (CT), as well as color and position of box to go behind text
- o Create pages (more than two lines) of computer generated text (PT)
- o Create frame branch files, to identify the GOTO when the sequence is interrupted
  - Interrupt motion (IM) files
  - Step thru (ST) files
  - Scan and branch (SB) files
- o Create random access (RA) file, to identify which of several branches (up to 20) the lesson will go to
- o Specification of touch areas for light pen activation
  - Standard touch areas
  - User designed touch areas
- o Specification of which responses are correct
- o Identification of start and ending frames for a series of stills (KB), plus the first frame to be displayed
- o Establish program link (PL) functions, which can be turned on and off in accordance with the lesson design
  - BACKUP - Review previous frame
  - EXIT - Allows user to exit
  - SUSPEND - Allows user to exit the program, then return to the same place at a later time
  - DEBUG - Superimposes RECORD ID, videodisc frame numbers, and graphic boxes representing touch coordinates on top of video pictures, as well as permitting access to any PMS record (event)
  - LAST DECISION - Returns user to previous decision point
  - PLACEMARK - Permits user to "mark" a record for easy access at some other point in time
  - TRACE - Outputs to hardcopy printer specific RECORD IDs accessed during lesson
  - DEFICIENCY REPORTS - Allows user to identify and annotate lesson or programming deficiencies to be printed at termination of lesson (NOTE: works only with interim-EIDS)
  - MENU - Allows user to return to the last menu accessed
  - MARGINAL NOTES - Allows user to add information, a note, which can be reviewed at a later time (NOTE: works only with interim-EIDS)
  - SOUND CUES - Provides aural cues to student, such as acceptable or unacceptable inputs (touch or keyboard)
- o Establish a loop counter (LC), used to inhibit student access to certain records, by branching to other locations
- o Specify requirement for multiple correct student responses (MC) prior to branching
- o Identify lesson locations to access, or hook (HK), non-PMS software
- o Permit linking of multiple subroutines (LS)
- o Identify a menu (MU), which has implications for program link functions
- o Past performance (PP) branches based on student performance on a given test, based on:
  - Time to respond
  - Weighted score based on time to respond
  - Point score
  - Combinations of these three
- o Record keeping functions
  - Test (TE) identifies a series of records that in total make up a test
  - The capability to define which records in a test require scoring (RR), i.e. which are test questions
  - Show the results (SR) of student performance to the student
  - Weighted time (WT) identifies a total time for a given sequence of records or events to specify a pass/fail criterion based on time
  - Record Time (RT) monitors the timing for a sequence of events, and keeps the data for later use

- o Access fill-in-the-blank (FB) capability (NOTE: works only with interim-EIDS) to turn off light pen and turn on keyboard to permit student to answer questions by entering text
- o Obtain personnel data (PD), name and/or rank and/or serial number (NOTE: works only with interim-EIDS)
- o User personnel data (UP), name and/or rank and/or serial number, obtained with PD, to customize a text screen display (NOTE: works only with interim-EIDS)
- o Deletion of production and post-production only records not needed for lesson execution

TESTING AND DEBUGGING

The process of testing and debugging is one of the most critical stages of interactive courseware authoring. Regardless of how well the rest of the process has been accomplished, failure at this stage can ruin the whole program.

IDEAL

The ideal system facilitates testing and debugging by including these characteristics:

- o Identification of syntax and/or format errors prior to lesson execution:
  - Automatic marking of specific location and type of error
  - Easy access to error location and annotation to facilitate correction
- o Run-time modification of computer generated test or graphics:
  - Automatic update of source code when modifications are made
  - Easy change of content, size, font, location, or color of text
  - Easy change of size, location, colors, or name of graphics
- o Run-time creation or modification of touch (actual touch, light pen, or mouse) responses:
  - Easy creation or modification of response location
  - Easy creation or modification of response size
- o Start at any place in lesson
- o Mark location of specific problem spots for later modification
- o Option to immediately terminate lesson execution and go to same location in source code
- o Run-time ability to view video, still or motion, anywhere on disc:
  - All normal audio and video commands available
  - Identify modifications required to video and audio commands, with automatic updates to source code
- o On-screen aides to assist instructor:
  - Full identification of record or event during run-time
  - Help function for error corrections, both run-time and off-line

PMS

To aid the programmer/developer in troubleshooting the lesson execution, there are programming functions and program utilities available. The programming functions include:

- o SUSPEND - allows user to exit the program, then return to the same place in the lesson at a later time
- o DEBUG - Superimposes RECORD ID, videodisc frame numbers, and graphic boxes representing touch coordinates on top of video pictures, as well as permitting access to any PMS record (event)
- o PLACEMARK - Permits user to "mark" a record for easy access at some other point in time
- o TRACE - Outputs to hardcopy printer specific RECORD IDs accessed during lesson
- o DEFICIENCY REPORTS - Allows user to identify and annotate lesson or programming deficiencies to be printed at termination of lesson (NOTE: works only with inter-EIDS)
- o MARGINAL NOTES - Allows user to add information, a note, which can be reviewed at a later time (NOTE: works only with interim-EIDS)

Specific utilities include:

- o Error checking during data entry, as incompatible programming functions are determined during the PMS stages when they are entered
- o Error Reporting Utility - This is a subset of the conversion utility, and can be used to spot check some errors prior to completion of the entire lesson, and thus reduce the amount of testing and debugging after the total lesson is complete
- o Conversion Utility - This is a five phase conversion process that continuously looks for errors to report, return to CPM (or DOS) when errors are found
  - Not every possible error will be found
  - Content errors, such as wrong video frame, text typos, location of graphics or touches, cannot be found by this or any other utility
  - Documentation does not tell what kind of errors will be found, or during which stage they will be found, although experience will quickly allow the user to identify them

RECORD KEEPING

As the concept behind interactive media is to provide the instructor with more time for other duties, automatic record keeping is essential. If the system provides useful, relevant data to the instructor, then instructor will accept and use the lessons. Failure of the system to provide these data makes the system unusable.

IDEAL

At a minimum, the record keeping functions of the system must include:

- o Student name, rank and ID number
- o Time spent on the lesson
- o Hardcopy printout of the records

In addition, there are other record keeping functions that should be available to the instructor. It may be desirable to make them instructor variables that can be turned on and off at the instructor's (or school or command) discretion. They include:

- o Student path through the lesson, assuming multiple paths are available
- o Specific errors made

- o Kind of error made, e.g. critical, major, minor
- o Correct responses by student
- o Anticipated correct response by student
- o Test results
- o Number of times a student accessed a lesson, or any part thereof
- o Complete history of all student actions
- o Performance indices; weighted scoring, comparison to pre-established norms, relative rankings, etc., both for a single lesson and for the entire curriculum

Further, data my be desired about the curriculum itself. Large groups of students may be having difficulties with the same part of a lesson or curriculum. The ideal system will assist instructors in identifying those trouble spots in order to allow modification and improvement of the instruction. Therefore, the system should be capable of generating cumulative data to identify:

- o Actual numbers and percentages of students answering specific test questions right or wrong, including breakdown of incorrect responses
- o Actual numbers and percentages of student selecting incorrect responses to non-test stimuli
- o Average and standard deviation performances by student population for each lesson or identified sub-set of a given lesson, for both errors and score

<u>PMS</u>

PMS incorporates a large amount of record keeping in the development of the lesson material. Almost any type of information about the project is available in multiple formats at any time during the development process. These capabilities are covered in detail in the paragraphs on the stages of development. The exhaustive organization and sorting possibilities available are highly commendable.

Record keeping within the realm of instruction is less comprehensive. There are basically two types of record keeping available within the lesson environment:

- o Test results which are recorded automatically according to the designer's specification of the available formats
- o Marginal notes and deficiency reports which are generated by exiting from the lesson via the program linked function capability

Also available is the initial input of personal data for identification.

USER FRIENDLINESS

Historically, user friendliness has been strictly defined as simplistic, menu-driven systems that require no intelligence to use. Unfortunately, that lack of intelligence is often carried over into the finished product. For the ideal authoring system, user friendliness should mean being easy to use, but not at the expense of being easy to learn. Another very important aspect of user friendliness is speed. Videodisc players have reached the ability to search to any frame in less than one second, but some authoring systems require several seconds, or longer, to accomplish even the simplest tasks.

<u>IDEAL</u>

English language structure will enhance the friendliness of the system, as well as eliminating strange abbreviations and codes. But the system must be able to accommodate the user who is capable of intelligent improvisation and who is usually frustrated by systems that will not provide flexibility. To meet these user requirements while retaining the better aspects of the easy to learn systems, several current methodologies are promising. One of these is the use of pull-down windows, so that the user can use a simple fill-in-the-blank routine or obtain whatever level of help is needed to complete the job. Simultaneously, the actual English language source code should appear on the screen so that the user can learn, by observation, easier ways to accomplish the same tasks. If no help is needed, the developer simply enters the necessary English language commands. The ideal authoring system must be fast so that the developer can be more efficient, and also so that the developer is not frustrated by having to continuously wait for the system to "do its thing."

<u>PMS</u>

There are three types of user friendliness that need to be addressed; those of the novice developer, the experienced developer, and, ultimately, the student. These do not necessarily overlap. In the first category, the following attributes are friendly:

- o PMS is a highly structured, clearly defined environment in which it is difficult to 'get lost' as long as you remember which phase you are in
- o There are relatively few choices available at any given juncture, making it easy to remember what is expected
- o Once memorized, the codes are simple to recall and use
- o VDB I is highly friendly in its entirety

In the second category, the experienced developer will find these features especially attractive:

- o Program Linked options, especially Suspend and Debug
- o Within the capability limitations, the ingenious developer can devise a number of nice effects without too much difficulty. The features that lend themselves to manipulation are:
  - Knob Turn
  - Loop Counter/Loop Reset
  - Scan and Branch
  - Step Through
  - Special Function

The student will find the following functions to be friendly:

- o Program linked functions including:
  - Suspend
  - Placemark

DISCUSSION

Again, the need for a standard is paramount. Whether that standard is PMS or some other system is less important than the identification of the standard. The truth is that no standard has been identified, and only PMS has been proposed as a universal system. In the comparison of PMS to an ideal system, PMS comes out well in some areas, but needs improvements in others. On the whole it is positive, but falls short of being a truly universal system. Its strengths are in the automated storyboarding and production areas, and its weaknesses are in hardware compatibility and programming flexibility. The following paragraphs identify those strengths and weaknesses.

HARDWARE

PMS is designed to be used with a single source of hardware, either Interim-EIDS or Production EIDS, depending on the version of PMS available. In this respect it is no better than some of the commercially available software packages designed to help sell a specific piece of hardware. In addition to the computer restrictions, the videodisc player, student input device (light pen), computer graphics and text capability, and audio storage medium are limited. PMS has no utility for those who have, or will acquire, different hardware. It does not maximize use of existing DoD hardware systems. While there is a linking function that permits unique software to be utilized in specific applications, PMS software is not capable of integrating or supporting any lesson material invoked by the "HOOK" function. However, attempts have been made to make EIDS more compatible, with an IBM compatible system, so that PMS will in effect have more universal utility. To be truly universal, however, PMS needs to expand its hardware horizon.

AUTOMATED STORYBOARDING

Probably more than any other available software package, PMS meets the need for automated storyboarding. As presently implemented, however, PMS needs to be upgraded, improved, in the following areas:

- o Documentation space within a single record or event needs to be increased
- o The extensive use of abbreviations and/or codes needs to be reduced
- o Art (stick figures) during the storyboarding process would be a major enhancement
- o Computer generated art and text should be viewable during the storyboard stage
- o More on-screen help should be available during this stage
- o Extensive pagination required to use system capabilities must be reduced

VIDEO PRODUCTION AND POST-PRODUCTION

PMS again is superior to most other available products in the area of production and post-production. The strengths are the ability to generate the paper products for production control and the ability to use the data base in support of production. Greater things are planned in this area, but at present there are some problem areas that could be addressed in future versions:

- o New video requirements added to the video data base (VDBI) during video production are not automatically integrated into the PMS data base. New data entered via VDB I can only be located and viewed in VDB I. The writer or other individual must go back into the storyboarding stage and manually add the relevant information.
- o No electronic interfaces are present. There are no automatic updates to the database, no way to communicate for automatic video text generation, and no automatic generation or control of the Edit Decision List.

LESSON PROGRAMMING

The ability of PMS to accomplish programming functions is somewhat limited. There are some good features, not common to other authoring programs. Especially notable are:

- o TIMED STILLS
- o KNOB TURN
- o INTERRUPT MOTION
- o SCAN AND BRANCH
- o STEP THRU
- o WEIGHTED TIME

On the other hand, the following programming limitations inhibit the ability of PMS to provide good instruction:

- o More than eight branches from a single record are not possible without artificial contrivances
- o Only contiguous videodisc frames or sequences (either still or motion, not both), can be used in a single record or event
- o A maximum of four graphics, less any other desired special features, can be used in a single record
- o One graphic cannot be superimposed on top of another, e.g. a circle on top of a square, unless the special HOOK function is used to call in a graphic created outside of PMS
- o Multiple events or records are often required to accomplish a single task or related group of tasks
- o Extensive menu pagination is required to change from "add" to "modify" mode, or vice versa
- o The system limits you to pre-existing capabilities, and changes are virtually impossible to obtain within the constraints of time associated with a lesson development project
- o Videodisc player capabilities are not fully utilized:
  - No reverse motion
  - Slow motion is a single speed, not variable
- o No capability exists for audio other than linear videodisc audio, tracks 1 and 2
- o "Seamless" delivery of lesson material is impossible - video or computer generated images cannot be automatically retained from one event or record to the next:
  - The video frame or computer text specified in each event or record must be re-displayed
  - Computer generated text or graphics must be re-generated for each record or event
  - Videodisc and computer generated image combination specification is restricted to which appears first; you cannot create computer text or graphics while viewing a video image, and you cannot search for a video frame while looking at a computer generated page
- o The only student interfaces possible with interim EIDS are the light pen and keyboard, but only the light pen will be available with production EIDS
- o Random selection works only in selecting the first event or record of a sequence, severely restricting testing options
- o Determination of correct and incorrect responses available only during testing
- o No discrimination between various kinds of wrong answers is possible
- o Only one right answer is acceptable per question when using weighted scoring
- o There is no branching between lessons based on performance

TESTING AND DEBUGGING

Some of the PMS utilities are designed to facilitate testing and debugging. When the execution code is assembled, certain errors are printed on the screen, which makes it easy to go back and fix the problem. Two major problems with this scheme were immediately obvious. There are five utilities or phases during assembly, and an error may not be found until the fifth phase. A lot of time will have transpired by then (between 20 and 30 minutes for a 50 record lesson). After fixing the problem, you must go through all five phases again, you cannot just rerun the failed phase. Another facet of this is that the assembly process is halted at the end of any phase in which an error is detected, so it isn't possible to go through the complete assembly process and then resolve all detected errors at once. The other problem is that it didn't catch every error, not even all format errors. These errors are not discovered until you attempt to run the lesson. One of the good ideas was to have the software identify any records or events that are never accessed within the lesson structure.

No run-time testing and debugging aids are available, except that some movement around the lesson is possible if you are clever. The documentation for such manipulations is sketchy, but it can be done. If the location of text, graphics, or touch is not correct, you must exit execution, go back into the PMS III, make your fix, reassemble the lesson, and try it out again. That can take a very long time. It takes so long that the developer is likely to feel that the lesson is good enough as it is and not bother to fix it. This attitude is also engendered by the process of verifying video frame numbers for start and stop of such good video commands as INTERRUPT MOTION, SCAN AND BRANCH, and STEP THROUGH. Close enough. The result is that the best features become just too much trouble to use properly.

RECORD KEEPING

In PMS, record keeping is minimal. The "TRACE" function allows an intelligent instructor to follow a student's path through the instruction, but it isn't easy. Some branching within a lesson based on student performance on a test is claimed, although it did not work as published when tested, but no capability is available outside of the test. Further, there is no ability to branch between lessons, or to carry performance across lessons. The lesson validation capabilities are non-existent. In addition, any data generated must be retrieved immediately. it cannot be stored in memory for later recall. If "TRACE" is not turned on prior to the lesson start, there is no way to activate it during the lesson.

USER FRIENDLINESS

An asset for PMS is that it is relatively easy to learn. Inexperienced users are trained in the total capabilities of the system in a one week workshop, and then can create lessons using the software. However, in that it does not provide flexibility, it is not easy to use. Lack of capabilities is one major cause, slowness is another.

CONCLUSION

PMS is a good start toward the ideal of a universal authoring system. However, while PMS may meet current and projected needs for Army Extension Courses, as presently configured it does not embody all of the characteristics of a universal authoring system. A very substantial amount of work is necessary to achieve that. PMS can be improved, and has the potential to meet the needs of the ideal system, and it the hope of all concerned that it will meet those needs. It certainly is superior to most competing systems in certain aspects such as storyboarding and production management.

One suggestion that has been forwarded is to perform similar analyses on other authoring systems to ascertain which is closest to meeting the needs of the universal system. There are two problems with that concept: there are too many systems that are currently in use, and all of the other systems are commercial products, and not under government control - no assurance that the changes necessary will be implement is available. To selectively sample from the existing systems opens the possibility of missing the one best system, and, as has been the case in previous such studies, would undoubtedly elicit all kinds of protests from those not sampled.

A second suggestion is to do nothing and let the market place decide which is best. This approach hasn't worked yet and is not likely to do so. The "best" system may be forced out of the market because of factors totally unrelated to overall quality and utility.

Until a standard is approved, controversy will continue as to whether or not this or that system is best. Compared to an ideal standard, PMS holds up as well. However, without a true standard, neither PMS or any other system will meet everyone's needs.

ABOUT THE AUTHORS

James R. Stonge is a Senior Instructional Systems Engineer currently supervising the courseware development activities of the Display/Trainer Products, Electronics Systems Division of Grumman. He is responsible for all technology associated with instructional systems development in that organization, as well as supervision of ISD personnel. Prior to assuming this responsibility he was ISD manager for AMTESS/MITS technology at Grumman. Prior to joining Grumman he was on the faculty of Indiana University. In addition to an AB, MBA, and MS in Instructional Technology, Mr. Stonge is completing his doctoral dissertation. He is a frequent contributor and presenter at I/ITSC and other national symposia.

Linda G. Jensen is an Instructional Systems Design Engineer in the Display/Trainer Products Group of Grumman's Electronics Systems Division. Her primary responsibility is E-2C training applications. She was also the ISD manager for Grumman's A-6E SWIP videodisc project. Her background includes nine years in development, modification, maintenance, and technical support of Air Force flight simulators. A graduate of the Army's PMS workshop at Fort Eustis, she wrote and presented a paper at the US Air Force Technology in Training and Education (TITE) conference on the need for a universal authoring system. She also attended the Training Technology workshop on videodisc applications at the invitation of the Department of Defense and the National Security Industrial Association.

THE USE OF A PART-TASK AIR INTERCEPT TRAINER IN F-16 AIRCREW TRAINING: RESEARCH RESULTS

George F. Lahey
David C. Hubbard
University of Dayton Research Institute
Dayton, Ohio

Bernell J. Edwards
Operations Training Division
Air Force Human Resources Laboratory
Williams Air Force Base, AZ

ABSTRACT

A training research program was initiated by the Air Force in 1984 to study applications of microcomputer technology to aircrew training. Emphasis was placed on the development of a methodology for identifying opportunities for part-task trainer applications and on demonstrations of the potential of part-task trainers built around microprocessor technology. Out of that program a number of part-task trainers have been developed and are being used to support pilot training programs. The most recent of these, the Air Intercept Trainer (AIT), is a low-cost, high fidelity, classroom device used for training F-16 air-to-air intercept skills. The intercept performance of experienced pilots converting from the F-106 to the F-16 who trained on the AIT was compared to the performance of others using classroom procedures. The AIT is described briefly and the results of the experiment are presented.

INTRODUCTION

The Air Intercept Trainer (AIT) provides a dynamic simulation of F-16 Head-Up Display and Radar Electro-Optical Display (HUD/REO) images in near real time. Ownship maneuver capabilities are provided using F-16 throttle and stick controllers as input devices. The dynamic effects of ownship maneuvers and target relative motion for single targets or for multiple targets, each with its own heading, altitude, and airspeed, are presented. The AIT was developed by the Link Flight Simulation Division of the Singer Company under contract to the Operations Training Division, Air Force Human Resources Laboratory, Williams AFB. The system input and output are managed by the 68020 microprocessor of a Motorola VME2000 computer programmed in Fortran. The HUD/REO displays are presented on commercially available CRTs. A photograph of the AIT is presented in Figure 1.

Figure 1. Air Intercept Trainer

The AIT incorporates an integrated Instructor/Operator Station (IOS) from which the instructor controls the intercept scenarios presented to the student. The IOS provides keyboard entry for menu selection of scenarios, the ability to freeze and unfreeze motion, and a display to present plan and gods-eye views of the intercept, that can be used for monitoring the intercept as well as for training purposes.

Target data can be presented on the radar in either freeze mode, in which both the target and ownship are frozen in space, or in dynamic mode, in which the targets move at constant speed on a constant heading. Individual exercises start with the system in the freeze mode. The instructor then controls system function by unfreezing or freezing the targets at any time during the intercept for instructional purposes. After target detection, the intercepts can be run in either flyable or non-flyable intercept modes. In flyable mode, ownship remains on freeze until lock-on, then is 'flown' by the student through the rest of the intercept. In the nonflyable mode, ownship stays on freeze to lock-on, then the computer takes over and runs a 1V1 intercept using an algorithm that reads the target's range, altitude and aspect angle and controls ownship accordingly.

Background

The initial model of the AIT was installed for research and demonstration purposes at the Arizona Air National Guard (AZ ANG) facility in Tucson. The initial model, located at Tucson throughout the research period, was nonflyable in that it had fixed throttle and stick settings. Only the radar control switches were active. Thus, while the student could operate the system for detection, all intercepts run in Tucson were run under computer control. In the second model, maintained at Williams AFB during the course of this research, the throttle and stick were active, so that intercepts could be run under either student control or computer control. A

repertoire of 26 scenarios using from one to five of 67 preprogrammed targets was included in the software delivered with the trainer.

The AZ ANG uses the trainer in three one-hour lessons to teach: interpretation of the HUD/REO symbology and switchology (all throttle and stick switches are completely functional), use of the radar and HUD for target detection, and intercept management. Prior to initiation of the experiment described herein, student use of the AIT in these lessons was observed over a period of several months. The objective was to determine how best to measure the performance of the subjects. A scoring system based on a model of an idealized intercept was developed. The scoring system could be used for grading inter- cepts in both the AIT and the Operational Flight Trainer (OFT) being used to prepare the subjects for their first intercept ride in the F-16. The scoring system could not be developed in time to provide scoring on the AIT, however, so it was incorporated in the OFT. Thus, the effect of the AIT training could be measured on the basis of transfer of training to the OFT.

APPROACH

The subjects for this experiment were experienced Air National Guard pilots in training in Tucson for conversion from flying the F-106 to flying the F-16. All pilots had prior intercept training and experience. The majority of the pilots were part-time Guard personnel, several with airline experience. Their military flight experience ranged from a low of just under 1000 hours to a high of over 7,000 hours. At the point of gathering the transfer data, all subjects had completed roughly four weeks of F-16 conversion training in Tucson and had had at least six training flights in the aircraft. The training program included a mixture of group lectures, self-paced slide-tape lessons, and one-on-one supervised part-task trainer use, plus conversion flights in an F-16B.

Because this was a field research opportunity in which the researchers were responsible to interfere with the training process as little as possible, the researchers did not exercise control of the syllabus, the content of the written materials used in the academic portions of the training, the selection of IPs for different training events, or the communications of instructors during lectures and during use of the AIT.

Procedure

Subjects were randomly assigned to one of four experimental groups prior to receiving any training on the AIT. All subjects then received symbology and switchology training on the AIT (AZ ANG Lesson HR-2). Following the switchology lesson, subjects in Group 1 had no further AIT training, but received a one-hour lecture on target detection, followed by a second one-hour lecture on intercepts in which they previewed videotaped sequences showing the eight research intercepts. Subjects in Groups 2, 3 and 4 received two supervised one-hour lessons on the AIT at Tucson in nonflyable mode, one on detecting targets (HR-3), and the second on running intercepts (HR-4).

After their initial intercept training, all subjects went to Williams AFB from Tucson to participate in the research. Upon arrival at Williams AFB, subjects in Groups 1 and 2 were briefed on the research procedures, then went directly to the OFT (the Training Effectiveness Research Facility (TERF) at Williams AFB) to run a series of eight research intercepts. Subjects in Groups 3 and 4 had additional supervised intercept training (approximately one hour) in the AIT on arriving at Williams AFB before running the research intercepts. Subjects in Group 3 got the additional training in non-flyable mode. Subjects in Group 4 got the additional training in flyable mode.

The number of subjects available was limited by the class size (13 subjects) and the number of classes (3) in the F-106 to F-16 conversion training program during FY 87. Four of the available subjects were dropped from the experiment; one because of an extremely negative attitude toward the research, one because he had been deliberately placed in Group 4 because of poor performance during the earlier parts of his training, and two because they lacked the pre-requisite symbology recognition skills. The resultant group sizes were ten subjects in Group 1, nine subjects in Group 2, seven subjects in Group 3, and nine subjects in Group 4.

Measurement

Training effectiveness was measured by comparing the performance of the subjects during intercepts run in the TERF. All subjects ran the same eight intercepts in sequence in the TERF, with no feedback given until completion of the last intercept. The first intercept was a straight through intercept. The remaining intercepts required a stern conversion, and included a low aspect/beam intercept (Intercept No. 2), three medium aspect/front-quarter intercepts (Intercepts 3, 4, and 5), and three high aspect/head-on intercepts (Intercepts 6, 7, and 8). The degree of difficulty increases as the aspect angle increases, with head-on intercepts being the most difficult. Mean scores on the head-on intercepts (Intercepts 6, 7, and 8) were selected as the test measure.

Performance was measured against a model of an idealized intercept. The data gathered by the computer included 30 measures of conformance to parameters of the idealized intercept. Scores assigned for each measure were summed to provide a single score for each intercept.

The student could earn a maximum of 1000 points on each intercept. Penalties for such errors as gimballing the radar, breaking lock, negative overtake during the conversion, failure to uncage the missile prior to launch, and excessive time to complete the intercept were subtracted from the intercept scores. Negative scores were possible and were in fact recorded by several subjects.

Class Differences

Because the AZ ANG course was a new course with no prior history, some changes between classes were to be expected. In the interval between Classes 04 and 05, the AZ ANG changed the training from running baseline intercepts to running air defense intercepts. The baseline

intercept requires going to a collision course immediately upon acquiring the target on all intercepts. The air defense intercept requires going for turning room immediately during front-quarter and head-on intercepts. This required a change in scoring. The points awarded for establishing and maintaining collision during baseline front-quarter and head-on intercepts (100 points) were transferred to the subtask of establishing and maintaining turning room (previously 200 points) during the air defense intercept, in order to retain the total number of points per intercept at 1000.

Between Classes 05 and 06, the Guard changed the interval between the AIT lessons. During Classes 04 and 05, the AIT lessons had been given on separate days, giving subjects time to assimilate the training. During Class 06, HR3 and HR-4 were presented on the same day.

To adjust for the changes in procedures between the three classes, a class variable was used in the data analysis. The class variable has the added benefit of adjusting for history.

Instructor Differences

Seventeen different IPs were used by the Guard for the one-on-one training and the conversion flights in Tucson. Five IPs were used for the AIT instruction at Williams AFB. The assignment of IPs to subjects for individual lessons was a function of the availability of the instructor and subject and hence was somewhat random, except for subjects in Group 1. Subjects in Group 1 might have been assigned any one of the IPs for the symbology/switchology lesson (HR-2), but had only one of two IPs for the target detection lesson (HR-3), and all had the same instructor for the intercept lesson (HR-4). Based on a preliminary analysis and on the fact that one instructor had been used for most of the Group 1 students in the target detection lesson and all of them in the intercept lesson, the instructor assignment for the symbology/switchology lesson was used as a blocking variable in the analysis of the data.

Hypotheses

The hypotheses being tested in the research were that, a) the performance of subjects given the initial AIT training would not differ significantly from that of subjects given the lecture and tape training, b) the performance of subjects given the added AIT training at Williams AFB would not differ significantly from that of subjects given no additional training, and c) the performance of subjects given the additional training in the flyable mode would not differ from the performance of subjects given the additional training in the nonflyable mode.

Data Analysis

The data were analyzed using a randomized block design with instructor assignment on HR-2 defining the blocks. Orthogonal contrasts were used to test the hypotheses. Because of the small n's and the high variability expected in the data, we selected a level of significance of .10 for all significance tests.

RESULTS

The mean scores obtained are presented in Table 1. Individual scores and group means are presented in Figure 2. The overall mean was 497. The median score for all subjects was 503. The median scores for the subjects in the individual groups were 306, 455, 542, and 610, respectively.

Table 1
Mean Intercept Performance Scores by Groups

Class	Group 1 Lect/Tape	Group 2 AIT	Group 3 AIT+(nf)	Group 4 AIT+(f)
No. 04	269	552	516	586
No. 05	193	504	690	608
No. 06	648	418	552	497
Overall Means	370	492	585	563

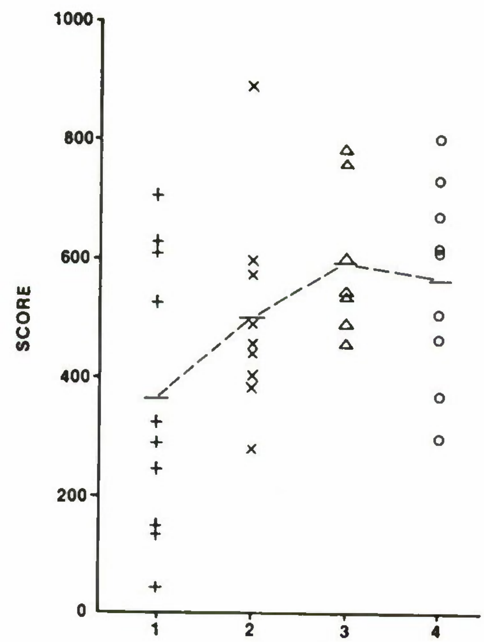


FIG. 2: INDIVIDUAL AND MEAN SCORES BY GROUP

There was no main effect for the instructor variable, so the instructor variance was pooled with the error variance. There was then a significant effect at the .031 level due to group membership, df=3,15, F=3.87. There was no class effect, but there was a significant interaction at the .022 level between group and class, df=6,15, F=3.52.

The difference in performance between the subjects in Group 1 and those in Group 2 was significant at the .090 level, df=1,15, F=3.28. The difference in performance between the subjects in Groups 1 and 2 and those in Groups 3 and 4 was significant at the .016 level, df=1,15, F=7.34. The performance difference between the subjects in Groups 3 and 4 was not significant.

The interaction was based principally upon a difference in the performance between Group 1 and Group 2 students in Classes 04 and 05 and those in Class 06, significant at the .090 level, df=1,15, F=13.27. There is a significant reversal of the relationship between the means, as is indicated in Figure 3. The performance of the Group 1 subjects in Class 06 was well above the performance of any other members of Group 1, and was above the 70th percentile in the overall data. By way of contrast, the performance of the Group 2 members of Class 06 was below the mean for the previous two classes, and at about the 33rd percentile in the overall data.

Contribution of Breaking Lock and Gimballing

The principal contributor to the variation in individual scores was the effect of breaking lock or gimballing the radar after target acquisition. The penalty for breaking lock without reacquiring within 20 seconds or for gimballing the radar was severe (-400 points), as it would be in the real world, and accounted for the lowest individual scores on every intercept. Scores on individual intercepts for subjects who broke lock or gimballed the radar ranged from -525 (for a student who gimballed, reacquired the target, then broke lock) to 315 (for a subject who broke lock early, then reacquired). The performance range for subjects who avoided breaking lock or gimballing the radar was from a low of 350 to a high of 945. The difference in frequency of breaking lock or gimballing was significant, df=3, Chi-square = 10.60, $p < .05$. Subjects in Group 1 gimballed the radar significantly more often than did subjects in Group 2; 12 gimbals versus 5, df=1, Chi-square = 3.23, $p < .10$. Subjects in Groups 1 and 2 gimballed the radar more frequently than subjects in Groups 3 and 4; 17 gimbals vs 5, df=1, Chi-square = 6.02, $p < .05$. The frequency of gimbals for each student is indicated in Table 2.

Table 2.
Gimbal Frequency for Each Student by Group

		% gimbals
Group 1:	1, 2, 0, 3, 2, 2, 2, 0, 0, 0	40.0
Group 2:	0, 0, 0, 0, 1, 2, 1, 1, 0	18.5
Group 3:	0, 0, 0, 0, 0, 1, 0	4.8
Group 4:	0, 0, 1, 0, 1, 1, 0, 0, 1	14.8

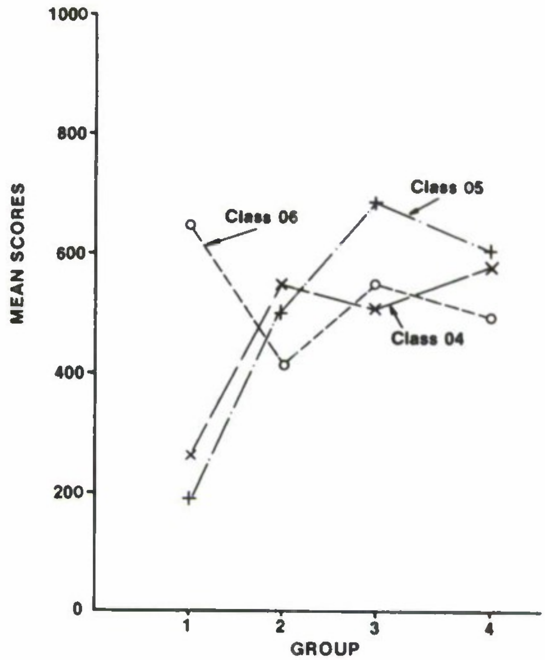


FIG 3: GROUP INTERACTIONS

DISCUSSION

As Figure 2 indicates, there was considerable variability in the scores. In general, the lower scores were the result of gimballing the radar. Subjects whose scores were above the median (503) did not gimbal the radar on any of the three test intercepts, while all 13 subjects whose scores were below 450 gimballed the radar at least once, and all six subjects with scores below 290 gimballed the radar two or more times. Gimballing the radar usually indicates that the subject was unable to interpret the symbols on the HUD/REO displays correctly. After target acquisition, the HUD/ REO displays give the subject all the information he needs to develop the intercept, provided he understands what he's seeing. Further, the onset of a situation that will lead to gimballing the radar is very apparent in the displays and can be corrected by proper aircraft control. Since the entry level skills of the subjects included extensive experience running intercepts, albeit in a different aircraft, failure to avoid gimballing is a substantial indication of a lack of understanding of the intercept geometry indications of the HUD/REO displays. The fact that the subjects trained via lecture and tape gimballed the radar more often than those who had had AIT experience, clearly indicates that the AIT makes a positive contribution to that understanding.

The absence of difference between the subjects in Groups 3 and 4 is surprising. This result suggests that simply viewing the computer-generated intercepts was more effective than

actually 'flying' them. One facet of the question, however, is that the nonflyable (computer-controlled) intercepts conformed to the intercept philosophy used in the research, so that the experience of the nonflyable group paralleled and reinforced their experience at Tucson, where the intercepts were also conducted in nonflyable mode. The subjects required to fly the intercept had to generate the intercept while at the same time learning to fly the AIT, a new learning experience. Because of the interference of the new learning, and to the extent that their performance in the AIT deviated from the prescribed intercept, they would have been less well prepared to perform research intercepts.

The performance of subjects in Groups 3 and 4 was little better than that of the subjects in Group 2, and, in fact, the subjects in Group 4 gimballed the radar almost as frequently as did the subjects in Group 2. Comparing the results of the experiment to the comments of instructors, subjects, and observers alike, who were of the opinion that the hands-on experience as a predecessor to going in the OFT was extremely beneficial, and even more specifically that the flyable mode was a significant contributor to understanding the intercept, raises a question as to what factor is at work in the experiment. One possibility, the interference of the new learning, was discussed in the last paragraph, but it doesn't apply to the Group 3 students. Their added practice should have helped. A study to control for all the variables except the added practice seems very much needed.

Following the completion of the research intercepts, all subjects had additional training in both the flyable AIT and the OFT. The uniform comment of the contract instructors who administered these lessons was that the pilots were performing intercepts very well by the time they finished these lessons. Although there were clearly individual differences in performance during these exercises, there was no evidence of any difference in performance as a residual effect of the research treatments used in the experiment.

CONCLUSIONS

It is concluded that the intercept training of conversion pilots is significantly enhanced by the use of the AIT. Individuals taught using the AIT develop superior intercept skills and are better prepared to run effective intercepts in the OFT. It is also concluded that the flyable mode of AIT training is not significantly superior to the nonflyable mode, albeit the opinions of instructors, subjects, and observers differed from the results. Data to eliminate this conflict between results and opinion are needed.

Additional data should be gathered to either confirm or modify the indications of the current research. It is recommended that data be gathered to determine whether, in fact, giving added training in the AIT when subjects come to Williams before putting them in the OFT improves the student's preparation. It is also recommended that data be gathered from pilots with no prior intercept training, to determine whether the AIT will have an even greater impact on training for those pilots who have had no previous experience in interpreting radars and visualizing the geometry of intercepts. It will also be desirable to discover whether the flyable feature of the AIT makes a significant difference when training the less experienced pilot.

Research counts now a number of successful part-task training devices built around microprocessor technology. Although the individual devices differ as to the skills taught, all have contributed or are currently contributing significantly to aircrew training. The proven ability of the AIT to enhance the acquisition of a higher level aircrew skill portends an expanded future for devices of this type.

ABOUT THE AUTHORS

Dr. Lahey is a Research Psychologist with the University of Dayton Research Institute, Flight Training Research Support Group, working at Williams AFB. He holds a doctorate in Educational Psychology, and has been doing training research for the Navy and for the Air Force over the past 18 years.

Dr. Hubbard is a Research Psychologist with the University of Dayton Research Institute, Flight Training Research Support Group, working at Williams AFB. He holds a doctorate in Experimental Psychology and the equivalent of a Masters in Applied Statistics. He has been doing training related research for the past nine years.

Dr. Edwards is a Research Psychologist with the Air Force Human Resources Laboratory group at Williams AFB. He holds a doctorate in Educational Psychology, and has been doing training research and analysis for the Air Force over the past 15 years.

TEAMWORK AND COMMUNICATION: A FORMULA FOR SUCCESSFUL DEVELOPMENT OF IVD

Sue Wiggins, Division Chief
Resident Training Division
United States Army Engineer School
Fort Belvoir, Virginia

Deborah Goodnow
Senior Education Specialist
PATRIOT Training Department
United States Army Air Defense Artillery School
Fort Bliss, Texas

Sandra Dee Kelley, Ed.D.
PATRIOT Project Manager
Diversified Technical Services, Inc.
6400 Convair
El Paso, Texas

ABSTRACT

Communication and interpersonal problems can create difficult situations in Government contracts. Unique ways in which these problems were handled on a PATRIOT missile system interactive videodisc (IVD) courseware development project at the United States Army Air Defense Artillery School (USAADASCH) are outlined. Identified problems and their potential impact on the project included: (1) Government versus Contractor attitudes, (2) top-down force resulting in bottom-up resistance, (3) lack of knowledge/understanding of the project, (4) time constraints, (5) red tape, (6) lack of experienced subject matter experts, (7) unavailability of validated Government-furnished materials, (8) rapidly changing equipment design, (9) several versions of equipment and documentation, (10) literal interpretation of contract, and (11) resistance to the project. The Government/Contractor management cohort developed a team approach as opposed to a "we-they" approach. Details and functional elements of this process are described. The end result was a quality product which enhanced the user's training capability and which was developed, delivered, and accepted on time and under budget.

INTRODUCTION

The United States Army Air Defense Artillery School (USAADASCH) is one of many military schools presently developing Interactive Video Disc (IVD) Courseware for technical skills training. Many industries consider IVD to be the training tool of the future, and the Government is no exception. The U.S. Army Training and Doctrine Command (TRADOC) has begun a major initiative to incorporate state-of-the-art instructional technology into every facet of the Army's training system. They anticipate that IVD will enhance the effectiveness of current instructional programs by increasing proficiency, by reducing training time, by minimizing safety hazards, and ultimately, by decreasing the high costs associated with producing technically skilled soldiers.

A contract completed last year for the PATRIOT missile system is one of the U.S. Army's IVD successes. The courseware is presently being used effectively to train PATRIOT maintenance and operational tasks in the classroom.

BACKGROUND

In 1984, the USAADASCH Directorate of Training and Doctrine (DOTD) was asked to explore the feasibility of developing IVD courseware for one of the newest and most sophisticated air defense weapon systems--the PATRIOT. The Government was about to field equipment which cost in excess of $150 million per system, but very few systems had been allocated for training purposes. Two highly effective training devices had been developed for PATRIOT, but they were not a panacea for the training problems since they were designed to teach only specific tasks. Hands-on training for the majority of critical tasks still had to rely on scarce tactical equipment.

The training concept of USAADASCH is that a soldier must be trained with the most cost-effective media. IVD simulates hands-on training by "demonstrating" a procedure and then having the student "practice" the procedure. By adding IVD instruction, the students would be familiar with the equipment geography and with the tasks to be performed prior to training on the actual equipment. The efficiency of tactical equipment training would be enhanced. Thus, IVD would fulfill the training concept and provide an attractive solution to PATRIOT's training dilemma.

DOTD envisioned a training strategy in which various media and methodologies would interact to form a solid instructional base. Platform instruction would be reinforced through corresponding IVD modules. IVD modules would, in turn, prepare students for training on the actual equipment. In short, IVD would serve as a bridge between the cognitive and psychomotor skills required to achieve total task proficiency.

INITIATION

The IVD project for PATRIOT was approved, and, with anticipation and some trepidation, the groundwork began. The Statement of Work (SOW) was approved, the Contracting Officer's Representative (COR) was appointed, and the contract was let. At this point, all resemblance to normal training development contracts ended. It soon became apparent just how challenging the project would become. IVD was new technology and required a whole new set of skills, knowledges, and attitudes. But reference materials were scarce, expertise was non-existent, and regulatory guidance had not yet been published. As the project forged ahead in the "learn-as-you-go" mode typical of emerging technologies, concomitant growing pains began to surface.

PROBLEMS

It seemed that the PATRIOT IVD contract was

written in terms broad enough to encompass "whatever" IVD turned out to be. Consequently, interpretation was nebulous. The Contractor looked to the Government to clarify their requirements. And, not being experienced in this field, the Government expected the Contractor to know what was needed. Attitude problems also arose when the Training Department, as end-user, was brought on board. The PATRIOT Department had just finished a two-year curriculum development effort. Now, they were being asked to revise their Program of Instruction (POI) to accommodate 90 hours of computerized courseware. Instructors were resistant because they didn't understand the alien concept of IVD, and they doubted the instructional validity of this new technique. Additionally, some were vaguely suspicious that their jobs might be in jeopardy.

As the contract progressed to the lesson development stage, a host of new problems emerged. Technical Manuals (TMs) changed so rapidly that it was nearly impossible for the Contractor or the Government Subject Matter Experts (SMEs) to keep pace with the revisions, let alone validate the new procedures. Often, technical errors in the TMs were not identified until after the lesson material had been written and approved.

Time, or a lack thereof, compounded the problems. The time allotted for the Government to review the draft courseware and verify its technical accuracy was inadequate at times—especially considering the volatile state of the TMs, the unfamiliar format of storyboards, and the heavy workloads already assigned the Government SMEs. The Contractor, who was committed to meet certain deadlines, also felt the constraint of time. Sketchy or erroneous documentation, limited Government SME support, and extended turn-arounds eroded their Contract Performance Plan (CPP) time.

The most frustrating problems, however, did not become evident until videotaping began. With such high demands on the equipment, scheduling and coordination had to be precisely orchestrated. Communication failures, along with equipment malfunctions, caused delays in this phase of the contract. More setbacks resulted when modifications to the equipment or TMs made approved lessons obsolete.

Of all the problems inherent to a contractual effort of this magnitude, the most serious were misconceptions and miscommunication. In the beginning, communication and coordination were effected on an "as needed" basis. Specific rules and responsibilities were not defined and the resulting "trial and error" method led to certain critical elements "falling through the cracks." This, in turn, led to frustration and hard feelings since everyone's best efforts were failing to produce the desired results. Conflict was inevitable considering these seemingly insurmountable problems, coupled with the diverse backgrounds and personalities of the individuals involved.

SOLUTIONS

The three managers representing the key organizations involved (and ultimately responsible for the success or failure of the contract) realized that measures had to be taken before the project died on the drawing board. After two or three brain-storming sessions which identified the majority of problems (both real and imagined), the decision was made to establish weekly working meetings to foster a "team" concept, to facilitate communication, and to expedite the development process.

These mandated meetings initially were met with resistance since they sometimes consumed two or three hours that could have been better spent "working." After a few sessions, however, attitudes began to change. Negative feelings and misunderstandings began to dissipate when all the team players could openly confront each other in a neutral, supportive environment. New respect was gained from the sharing of ideas, and confidence grew as accomplishments escalated.

The weekly working meetings did more than provide an open forum to air problems. They added structure, clarified responsibilities, and streamlined the coordination process.

Over a period of time, the team devised a set of routine operational procedures which considerably eased the burden for all. Now, before courseware is drafted, a strategy meeting is set up between the writer and SME. They carefully plan the design and flow of each lesson. The important task procedures are identified, flowcharted, and tracked through the TM. The level of difficulty, amount of interaction, and all pertinent data are recorded on the Strategy Outline Form shown at Appendix A. Thus, a document signed by both parties serves as a guide and as an instrument of improved communication.

Next, the task procedures and TM paths outlined during the strategy meeting are tried out on the equipment to ensure a valid, workable lesson. If at all possible, the writer/SME team that designed the strategy conduct this walk-through together. Many of the kinds of errors which previously were not detected until videotaping are now identified and corrected before development begins.

After a lesson is written, Government SMEs review the storyboards and suggest technical and/or design revisions which they think are appropriate. The turn-around time for this step was greatly reduced when the Government SMEs began recording their comments on the Change Record Form shown at Appendix B. The Contractor explains the revision process on the Errata sheet at Appendix C. The Errata Sheet and the affected storyboards are then resubmitted to the Government for approval.

Coordination of videotaping requirements was one of the most difficult hurdles to overcome. Appendix D contains an example of the "Request for Videotaping" letter which the Contractor now submits to the COR for each lesson. The letter identifies the task, the procedural steps to be taped, the desired videotaping dates, the condition and set-up of the tactical equipment required, ancillary tools and test equipment needed, and the number of demonstrators required. The letter is disseminated to the organizational elements involved and serves as the master plan for the coordination of videotaping.

Once the videotape is edited to conform to the approved storyboards, the Contractor's writers and SMEs and the Government COR and SMEs review the premaster tape. All appropriate revisions are incorporated before the master tape is sent for manufacture of the proof/validation disc.

After a complete IVD lesson has been converted to a proof disc and the software has been programmed, the review/revision cycle is repeated. Since careful planning and comprehensive quality control procedures have been executed, it is rare that serious problems surface this late in the developmental phase.

While the policies and procedures described were successful in alleviating most of the problems with this contract, not all could be solved so readily. Fortunately, the team approach nurtured the flexible, cooperative attitudes needed to solve some of the more complex problems such as

apprehension, lack of education, and constantly changing requirements. The Government and the Contractor arranged demonstrations, tours, and workshops to educate, enlighten, and involve personnel.

The contract spelled out time lines and specifics for the review cycle that could have resulted in contract slow-downs, out-dated lessons, and poor quality products. Revision of courseware materials is one example. Everyone cooperated in revising and updating courseware until the latest possible date regardless of the milestones specified in the contract.

CONCLUSION

Completion of the first PATRIOT IVD project shows that a plan can come together despite what seem to be insurmountable problems. Ninety hours of IVD Courseware have been incorporated into the 24T POI, and approximately 180 additional hours are currently being developed.

The success of the project can be directly attributed to the "Team Concept" and dedication of all involved. A cooperative, effective system has been instituted to design instructional strategy, validate training, and evaluate the quality of every phase of the project.

In short, from a chaotic and disjointed beginning, there emerged a committed professional team which transformed the project from "management by crisis" to a finely orchestrated and well-planned endeavor. This effort resulted in a mutually acceptable quality product which was developed, delivered, and approved on time and under budget.

ABOUT THE AUTHORS

SUE WIGGINS is the Division Chief, Resident Training Division, U.S. Army Engineer School, Fort Belvoir, Virginia. Her previous position was Chief, Applied Technology Branch, New Systems Training Office, Directorate of Training and Doctrine at the United States Army Air Defense Artillery School at Fort Bliss, Texas. Sue, who has a B.S. in Management, was the COR for this project. She has worked for the Government for 16 years.

DEBORAH GOODNOW is the Senior Education Specialist for the PATRIOT Training Department at the United States Army Air Defense Artillery School at Fort Bliss, Texas. Deborah, who served as the project coordinator for the Government end-user, has a B.A. in Psychology and 5 years experience in Training Development and Instructional Technology.

SANDRA DEE KELLEY is the PATRIOT Project Manager for Diversified Technical Services, Incorporated, El Paso, Texas. Sandra has an Ed.D. in Curriculum and Instruction from New Mexico State University, Las Cruces, New Mexico. She was the Contractor's manager on the first PATRIOT project and is currently managing a second, larger PATRIOT contract for DTSI.

STRATEGY OUTLINE

TASK NO. 441-083-1158 TASK TITLE: 441-083-1158

TASK VALIDATED: ___ YES ________ (Date)
___ NO REASON: ________

REFERENCE(S):
TM 9-1440-600-20-1, with Change 5, paragraph 3-11.
TM ________, with Change ____, paragraph ____.
TM ________, with Change ____, paragraph ____.

LESSON OVERVIEW (Major Teaching Points): Test students ability to perform para 3-11 using gaming technique.

INITIATING SITUATION (Cue): Simulation - Missile Ready light is on when restraint pin is locked.

Post Test - Missile Ready light is off when is should be ON!

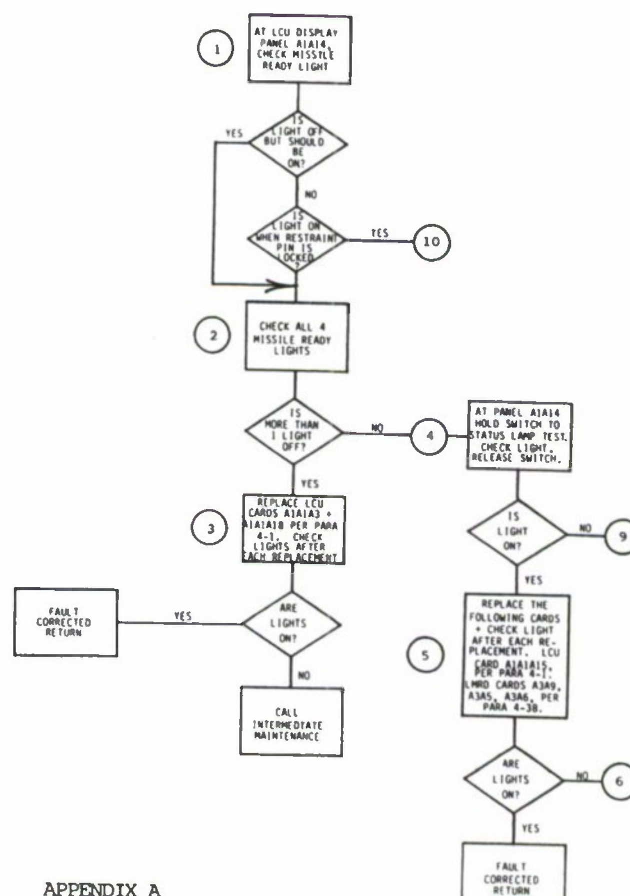
ENABLING/PRELIMINARY SKILLS REQUIRED: Question on Power Up and Down.

PATH (Steps): See Flow Chart.

DIFFICULTY LEVEL (Circle): (Entry) (Average) (Advanced)

SPECIAL NOTES: Gaming technique - see enclosure.

Author ________ (Signature/Date) Gov Rep ________ (Signature/Date)



APPENDIX A

Recommended changes to PATRIOT CAI lesson material

FROM: ______________________ DATE: ____________

____

TASK: ______________________

Frame	Line	Recommendations
1	3	(ex. TISK should be TASK)

APPENDIX B

ERRATA SHEET MM 8/14/87

LESSON: 441-083-1059

REMOVE FRAMES: 1120, 3500

INSERT FRAMES (NEW): 1120
(MODIFIED) 3500, 3650

CHANGE FRAME	GOTO TO READ	CHANGE FRAME	GOTO TO READ

APPENDIX C

TECHNICAL SERVICES INC.

P.O. Box 31820 • El Paso, Texas 79931 • (915) 779-6487

4 August 1987

Commandant
U.S. Army Air Defense Artillery School
ATTN: ATSA-DTC-ET (Mr. Joe Cabral)
Fort Bliss, Texas 79916-7090

SUBJECT: Request For Videotape Footage, Task 441-083-1107

REFERENCE: Contract No. DABT60-85-C-0573

Dear Mr. Cabral:

1. In accordance with Event 10, above reference, request eight hours of equipment and personnel time be scheduled anytime during the period 9-15 August 1987 for the purpose of obtaining videotape footage in support of Task 441-083-1107 (RS).

2. Equipment, personnel, and tool requirements are listed in Enclosure 1.

3. The following maintenance procedure will be performed: TM 9-1430-601-20-1, para 3-66, Radar Status Control--Indicator Panel A112 Status--Air Flow Fault--Curbside--LLE--Low Fault Isolation.

4. Please contact Mr. Fedler or me if you have any questions.

Sincerely,

Sandra Dee Kelley

Sandra Dee Kelley, Ed.D.
Project Manager

SDK:la

1 Encl
as

Enclosure 1 (Equipment, Personnel, and Tool Requirements) for

Task 441-083-1107

1. Equipment:
   a. Fully Operational Radar Station.
   b. Multimeter, Digital.
2. Personnel: 2
3. Tools:
   a. Screwdriver, Flat Tip, 1/4-In. Wide, 4-In.
   b. Screwdriver, Cross Tip, No. 2, 8-In.
   c. Switch Assembly, AC Line Test
   d. Cable Assembly, W150
   e. Cable Assembly, W106
   f. Cable Assembly, W101
   g. Pliers, Electrical Connector
   h. Wrench, Box/Open End, 9/16-In.
   i. (E)[1] Adapter W150A1

APPENDIX D

ROLE OF HUMAN ENGINEERING IN ADAPTIVE INFORMATION DESIGN FOR INSTRUCTORS

Mary A. Kanarian, Ph.D.
Raytheon Company, Submarine Signal Division
Portsmouth, R.I. 02871-1087

ABSTRACT

This paper documents the role of Human Engineering in the design of a "user friendly" instructor interface for a new generation On-Board Trainer Control Console (TCC). This role includes assisting in identifying program specific functions required, designing display and control formats to support the identified requirements and documenting display and control performance requirements in a manner that facilitates communications among design personnel (e.g., Systems, Software, Test) and to the customer. Types of human engineering analyses that are performed to identify program specific functional requirements are discussed. Display information and control presentation styles and feedback techniques (e.g., prompts, range cues, windows, inverse video) used to incorporate the customer specified requirements for the TCC into the instructor's display and control formats are outlined. The design techniques used are presented along with the rationale for their use. Additionally, tools that have been developed to accomplish the documentation task effectively (e.g., Switch Control Trees) are described.

INTRODUCTION

The purpose of this paper is to describe the role of human engineering (HE) in the design and development of a "user friendly" instructor interface for a new generation On-Board Trainer Control Console (TCC). The unit described is currently being used for a variety of trainer applications (e.g., on-board surface and subsurface sonar trainers and a shore-based sonar maintenance trainer). The "human engineered" features of the TCC offer the instructor the following benefits:

1. easy to use displays and controls which support all of the required instructor tasks and functions for Scenario Generation, Training and Test modes of operation,
2. a simplified, adaptive control structure which reduces opportunities for error and minimizes the training time required for the instructor to learn how to use the TCC, and
3. preprogrammed scenarios and innovative scenario modification features that minimize the required instructor workload at the TCC during scenario generation and training and increase the time the instructor can devote to trainee performance evaluation activities.

In the following sections of this paper, both the tasks performed by HE to ensure a "user friendly" instructor interface for the TCC and the "human engineered" features incorporated into the design to meet this goal are highlighted. First, the Instructor interface, the TCC, is described. Second, the analyses performed by HE to identify program specific instructor tasks to be performed and functions to be supported are detailed. Third, the major HE design requirements for the development of instructor display/softkey control formats for the TCC are outlined. Fourth, the display information presentation and control styles and feedback techniques that were used to meet these HE design requirements are presented. Fifth, documentation generation and communication tools developed to relate the interactive displays to the softkeys are described.

TCC DESCRIPTION

The TCC, shown in Figure 1, is a modular unit which is suitable for desk or bulkhead mounting. The unit includes a 14.02 x 14.02 Inch (35.61 x 35.61 cm) plasma panel with a touch sensitive bezel for sensing control and cursor locations, 1024 x 1024 pixel resolution and graphics capability. The plasma panel is used to provide interactive displays and softkey controls for generating and playing out complex training scenarios and conducting off-line system testing.

Figure 1. *Trainer Control Console (TCC)*

To use the TCC, the instructor touches the plasma panel surface at the spot where either the softkey control to be selected is located or the cursor is to be positioned. As the instructor's finger approaches the plasma panel surface, infrared beams located in the bezel of the plasma panel are interrupted and the X,Y coordinates of the desired softkey control location are sensed. The X,Y coordinate data is interpreted by the system and the plasma panel recongfigures to present the instructor with the logical display and softkey control options for completing the selected function.

The interactive display/softkey control formats have orange characters and graphics on a black background. The display brightness is discretely adjustable via a dedicated control on the front panel. The software for the instructor interface is modular in design to facilitate modifications to meet new user requirements at minimal cost.

INTERFACE DEFINITION TASKS

In order for the interface to be an effective work station, the plasma panel displays and controls provided must support all of the required instructor tasks to be performed. For complex training systems, the displays and controls provided need to be presented in a clear, logical manner to reduce the apparent complexity of the interface. Extensive, ongoing HE analyses are required to identify both the display information elements and softkey function control options needed and all of the conditional relationships between them. Additionally, HE must communicate the detailed display and softkey control performance requirements to other design personnel and to the customer to ensure that they are all included in the final design. Therefore, a first step in defining the instructor interface is to devise a cohesive plan for HE involvement in Instructional System Development (ISD) activities throughout the entire program.

HE Program Plan

For the trainer programs in which the TCC has been used, the responsible Human Factors Engineer initially defined the HE role in instructor interface design for each program by generating a program specific HE Program Plan (HEPP). The guidelines for generating a HEPP presented by the government in the Data Item Description document DI-H-7051[1] were used to determine the required areas of HE involvement. The guidelines for tailoring the HEPP for specific applications specified in Appendix A of MIL-H-46855[2] were used to determine the ISD tasks to be performed.

The HEPP is ultimately important to the user of the TCC because it describes the entire HE plan for ensuring that the trainer will be effectively and safely manned, operated and maintained. The HEPP describes the entire HE program, identifies its elements and explains how the elements will be managed. Additionally, it describes how the HE effort will be integrated within the total project and presents specific information about how and when the HE performance and design requirements specified for the trainer program will be satisfied.

Guidelines For HE Involvement In Instructional System Development Activities. It is important that the HE ISD activities included in the HEPP are both appropriate for the system being developed and adequate for ensuring the integrity of the user interface. The HE tasks which are detailed in MIL-H-46855 have been summarized, amplified and related to traditional systems engineering tools by Hiss[3] and by Rizy[4] as shown in the HE Development Process Chart presented in Figure 2. (The referenced boxes in Figure 2 are from MIL-H-46855. The unnumbered boxes were adapted from Hiss and Rizy).

Following the guidance provided in Appendix A of MIL-H-46855 and by Hiss and Rizy, tasks included in the HEPP encompass all phases of trainer system development. HE efforts are initiated with the preparation of the proposal and extend beyond the test and evaluation and final documentation and communication tasks outlined in MIL-H-46855. Additionally, HE will be involved in the teaching of the required operator course for the trainer programs.

The tasks typically contained in the HEPP to identify instructor interface requirements for the trainer programs include, but are not limited to, the following:

1. identifying relevant HE design requirements (e.g., those in MIL-STD-1472[5]) specified for the instructor interface,
2. conducting Mission Analysis developed from a baseline scenario to identify needed system functions; identifying present system user conventions through consultations with system area experts, operators and instructors,
3. assisting in the allocation of the identified new functions to either the instructor, hardware, software or some combination thereof,
4. conducting Functional Requirements Analysis to identify mission specific display and control functions needed,
5. designing display and control formats and structures to meet the identified instructor information and action requirements; defining Operational Procedures,
6. generating graphics of the proposed display and control formats using data from an existing scenario and then conducting a Task Analysis to ensure that the designs support the identified user requirements,
7. documenting display and control performance requirements in a manner that facilitates communications among both in-house personnel (e.g., Systems, Software, Test and Evaluation, Technical Publications and Integrated Logistics Support) and to the customer to ensure that the trainer system constructed incorporates the design features proposed by HE for the instructor interface, and
8. acting as a consultant on further issues concerning the operator interface, revealed as the design matures, for the duration of the program.

INTERFACE DESIGN REQUIREMENTS

The major HE design requirements specified for the TCC instructor interface by the customer for the program in which the TCC was first used were to:

1. incorporate relevant MIL-STD-1472 HE design requirements,
2. minimize required instructor training needed to learn how to use the TCC,
3. minimize instructor workload required during the conduct of training, and
4. develop a modular plasma panel display/softkey format design approach that could be easily adapted to accommodate either updates or new trainer applications.

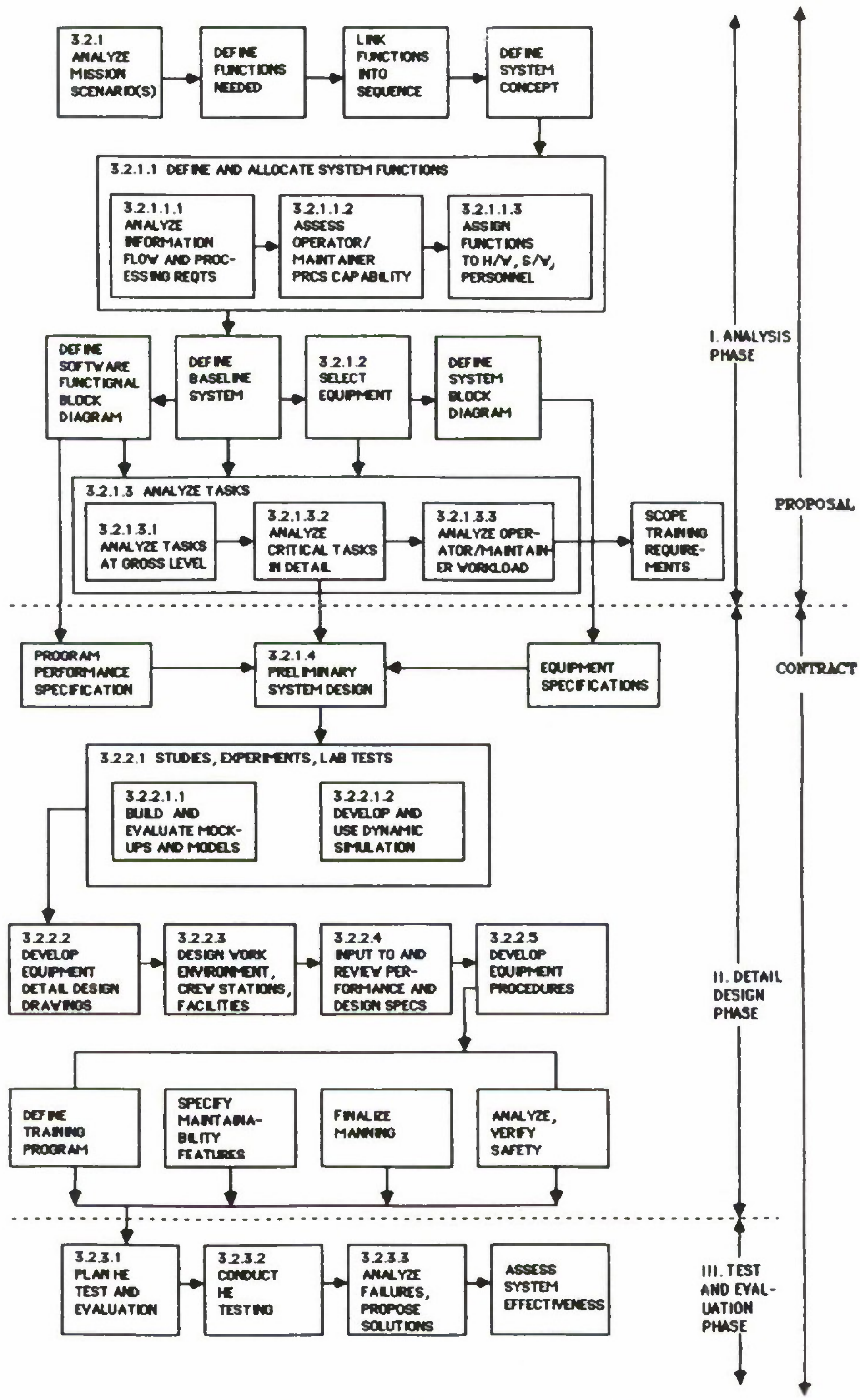


Figure 2. *Human Engineering Development Process Chart*

(The referenced boxes in Figure 2 are from MIL-H-46855. The unnumbered boxes were adapted from Hiss and Risy).

These four design requirements are of ultimate importance to both the instructor using the TCC and to the trainees. They are important to the instructor because they specify "user friendly" display and control characteristics, such as: readable display character and symbol sizes, functionally grouped and logically sequenced display information elements and controls, descriptive prompts and appropriate operator feedback. They are important to the trainee because they specify time saving features, such as preprogrammed scenarios, which reduce the time the instructor needs to interact with the TCC during the conduct of training and increase the time available for the instructor to devote to trainee performance evaluation activities. A summary of some of the display and control features incorporated into the TCC design to meet each of the four requirements is provided below.

METHODS USED TO INCORPORATE DESIGN REQUIREMENTS

MIL-STD-1472 Requirements

Functional Grouping. Display information elements and softkeys identified through the Functional Requirements Analysis mentioned above were grouped (e.g., OWNSHIP, OCEAN, etc.) to minimize instructor search time.

Logical Sequencing. Display information elements and softkeys were presented in a logical sequence, according to user conventions (e.g., OS MOTION, COURSE, SPEED, etc.) to minimize the time needed to learn how to execute a function.

Prompts, Range Cues. Descriptive system messages and operator prompts (e.g., DISK NOT LOADED) were also provided, as appropriate, to guide the instructor through the tasks at hand. Range cues, with software boundary checking, and preview data were provided, as needed, with the keypad to prevent illegal entries and to minimize data entry errors.

Positive Feedback. Positive feedback (e.g., inverse video presentation of softkey legend) and system messages (e.g., HARD COPY BUSY) were provided to indicate operator selections or system component state.

Requirement To Minimize Instructor Training

Design Simplicity. The display information elements and softkeys were presented in the same sequence on each display where they were included and the same display/softkey formats were used for both Scenario Generation and Training modes to minimize the potential for instructor confusion, cost of software and the instructor training time required. Additionally, touch control of the cursor is provided to simplify GEOSIT CONTROL activities (e.g., changing map center, hooking contact).

Requirement To Minimize Instructor Workload

Adaptive Controls. Displays, softkeys, and indicators were only made available as needed (e.g., OS MOTION SOURCE not changeable/available during training run) to prevent illegal entries. Lower-level softkeys that are used to further define a selected function (e.g., LOW, MODERATE, HIGH or the keypad) were only made available when the related top-level function softkey (e.g., NOISE LEVEL or COURSE) is selected to reduce decision making tasks. The management of the softkeys to prevent illegal entries reduces the apparent complexity and visual clutter of the instructor interface and reduces instructor workload.

Data Access Features. An information access window and softkeys to access related displays or the next higher display in the tree were provided, where appropriate, to minimize required instructor keystrokes and paging to gain relevant information.

Protected Controls. To reduce error from accidental activation of softkey selections which cause drastic changes in either trainer state or scenario state, all such function softkeys (e.g., EXIT MODE, FREEZE) require confirmation by also selecting the ENTER softkey provided with a prompt (e.g., PUSH ENTER TO FREEZE SCENARIO). The instructor can cancel an erroneous selection by selecting either another top-level softkey or the CLEAR softkey provided along with the ENTER softkey.

Preprogrammed Scenarios. Preprogrammed scenarios designed to meet specific Training Objectives can be used to conduct an entire training session with limited instructor intervention so that the instructor can concentrate on evaluating trainee performance. The instructor can modify an existing scenario or build one from scratch to meet the Training Objectives. For instructor convenience, the preprogrammed changes for the scenario generated are shown on a SCENARIO SCRIPT menu (see example in Figure 3).

Requirement For Modular Design Style

Two modular display/softkey control format styles, which have evolved over five years of research and experience designing display and control formats for plasma panels, were used in the designs for the TCC. The two styles selected were chosen because they are easy to use and because they easily accommodate either updates for adding new functions or for generating new displays for other trainer applications.

The graphic presented in Figure 4 shows the style used for all top-level "control only" displays.
The "control only" display style is used to guide the operator through mode, submode and menu option selection tasks (e.g., AAAAA (program name) OPTIONS, DISPLAY INDEX).

The graphic presented in Figure 5 shows the style of the general format used for all lower-level displays and the position of all general display/softkey features, which are provided as needed.

1. event/problem/test time, for use while either constructing a scenario, monitoring a scenario, testing the system or recording the time of scenario events of interest via the HARD COPY control;
2. selected trainer mode of operation, denotes the selected mode of trainer operations: SCENARIO GENERATION, TRAINING or TEST;
3. sensor status available/selected indicators, denote which sensors are currently turned ON in the scenario;

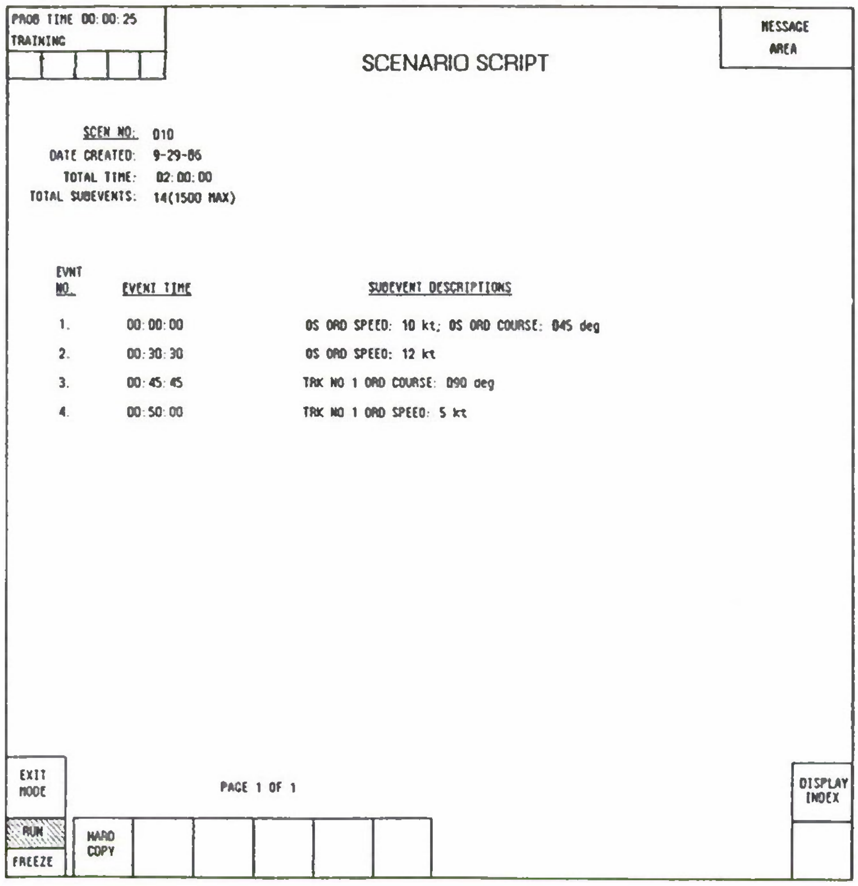


Figure 3. *Scenario Script Menu*

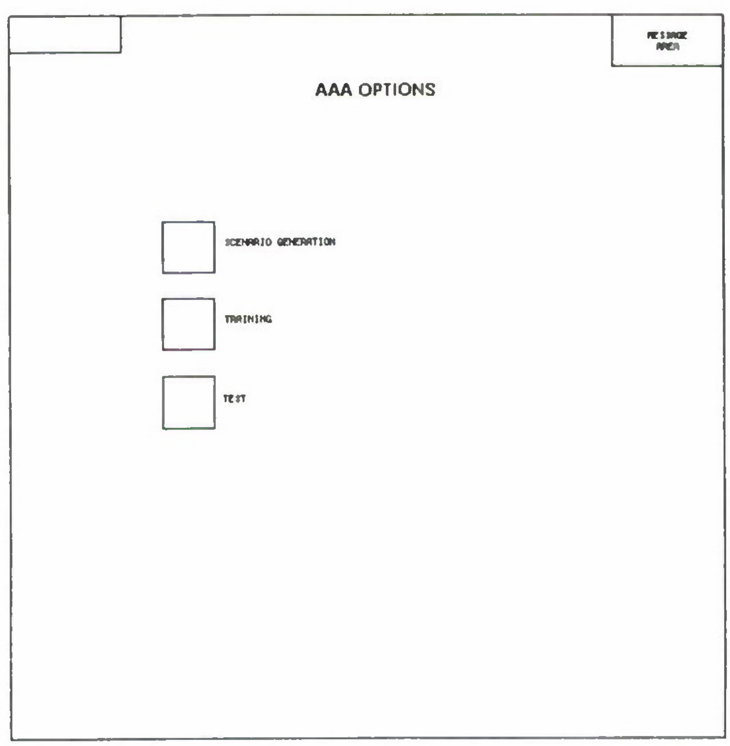


Figure 4. *Options Displayed At Power On*

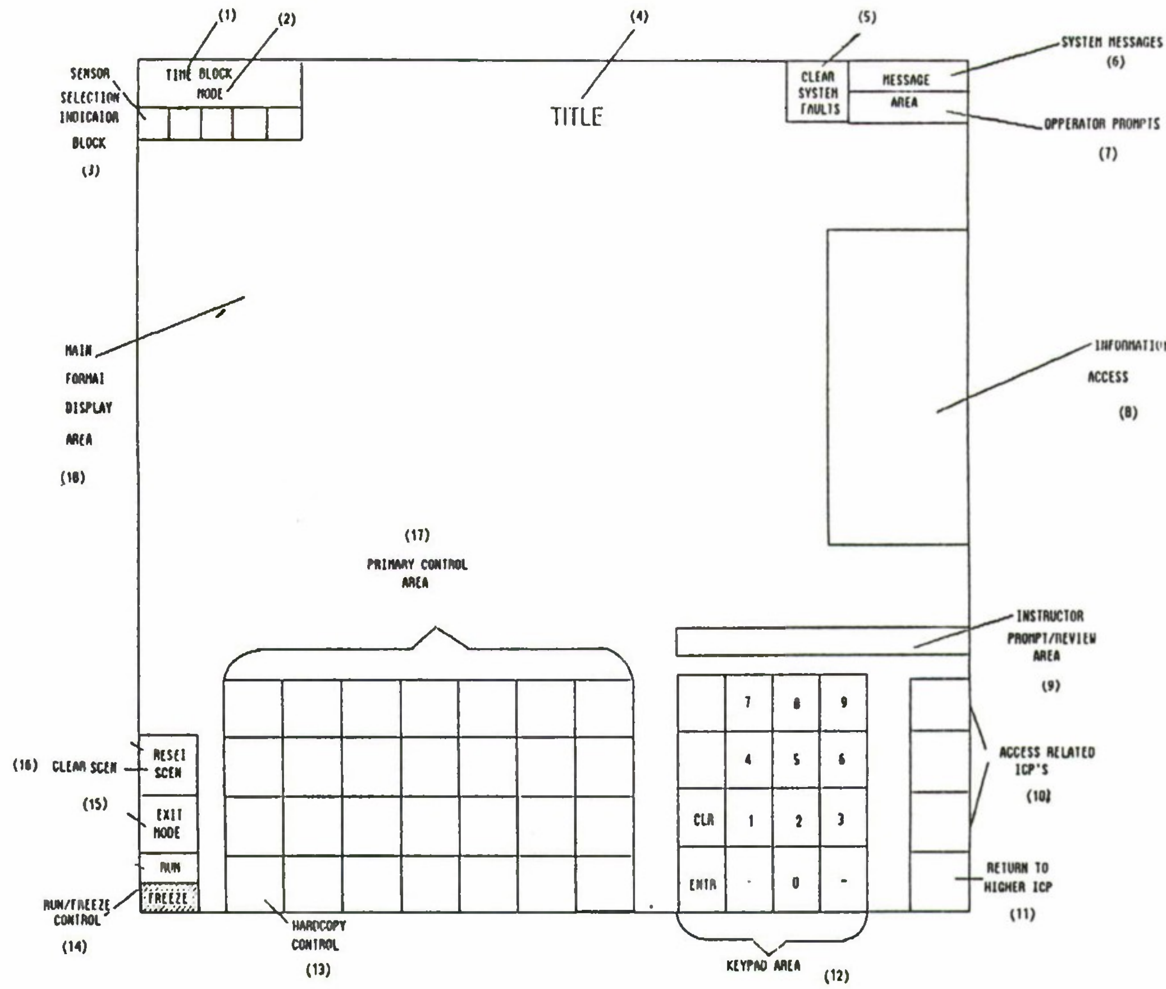


Figure 5. *General Display Format Features*

4. displayed format title, indicates the functional grouping of the display information elements and softkeys located on the selected page;
5. softkey to clear fault messages allows the operator to acknowledge and clear system faults;
6. area for system messages, informs the operator when system faults have been identified;
7. area for operator prompts (e.g., DISK NOT LOADED), help guide the instructor through selected functions;
8. Information Access Area window, presents relevant amplifying information;
9. range cue/data entry preview area presented with the keyboard, shows boundaries for legal entries for the selected function and echoes data as it is entered for instructor preview and verification;
10. softkeys to access relevant displays on same or different levels of the display hierarchy, reduce paging requirements to access related menus;
11. softkey to return the next higher level display in the tree (e.g., DISPLAY INDEX), facilitates accessing menus needed or returning to the previous menu;
12. area where the keypad is presented when numeric data entry is required to complete definition of the selected function;
13. HARD COPY softkey, enables the recording of critical scenario events for trainee debrief;
14. problem RUN/FREEZE softkey, allows the stopping and restarting the scenario during training;
15. EXIT MODE softkey, provides quick exit from the selected trainer mode of operation and releases equipment being used for training for Ownship use;
16. RESET SCENARIO allows for restarting the scenario at problem time 00:00:00 after scenario FREEZE; CLEAR SCENARIO allows for clearing the scenario from memory for conducting a free-play exercise for part-task training.
17. primary control area for function softkeys on lower-level displays, and
18. main format area for display of information and graphics on lower-level displays.

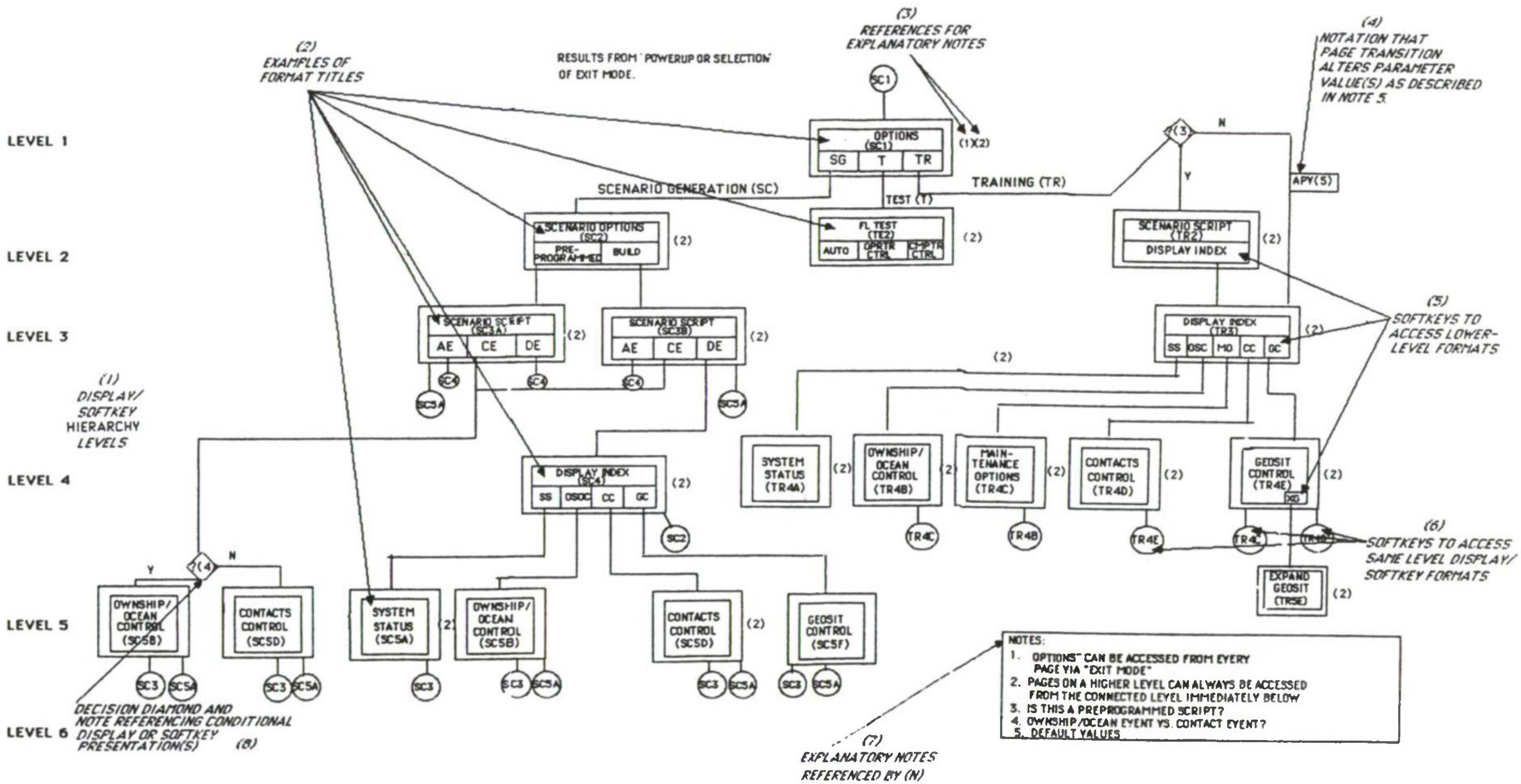


Figure 6. *Page Control Tree (PCT)*

DOCUMENTATION AND COMMUNICATION TOOLS USED

The interactive nature of the plasma panel display/softkey control formats simplifies the instructor interface, as noted above. However, it increases the complexity of the interface requirement descriptions and, consequently, the HE documentation and communication tasks. The document generation and communication tools refined or developed to relate the display/softkey requirements efficiently to in-house efforts as well as to the customer include: Display/Softkey Format graphics, Page Control Trees, and Switch Control Trees.

Display/Softkey Format Graphics.

Display/softkey graphics are generated on a computer for inclusion in documents (e.g., Computer Program Performance Specification, Test Procedure Manual), for use as mock-ups for conducting Task Analysis and for HE use at in-house and customer design review. Display/softkey graphics are developed for the top-level and lower-level general formats (as in the examples shown in Figures 4 and 5 above) and for each unique page of the display/softkey format set. Data from a representative scenario is used for developing the graphics of the display/softkey formats to demonstrate the operability of the proposed designs. An example of a SCENARIO SCRIPT display/softkey format containing scenario data is presented above in Figure 3.

Control Trees

Page Control Trees. The tool to explain the display hierarchy is called the Page Control Tree. The complete details concerning this display information documentation tool is contained in the guidelines for constructing Switch Control Trees (SCT's) developed by Clifford, DeFanti, Kanarian, Manning and Rizy([6]). The annotated example of a Page Control Tree, shown in Figure 6, summarizes the following major features of this tool:

1. hierarchy of displays
2. format page titles
3. references for explanatory notes
4. notation if parameter value changes occur with page transition (e.g., default values inserted)
5. softkeys to access lower level formats
6. softkeys to access same level display/softkey page formats
7. explanatory notes referenced by numbers
8. decision diamond and note(s) referencing conditional display or softkey presentation(s)

Switch Control Trees. The tool developed to explain the softkey presentations and their link to interactive display elements is called the Switch Control Tree (SCT). The complete details concerning this display information documentation tool is contained in the document outlining guidelines for constructing SCT's developed by Clifford et al. The annotated example of a top-level SCT, shown in Figure 7, defines information that can be presented on a switch or indicator by the following:

1. location of softkey/indicator on format
2. softkey/indicator legend
3. callup conditions
4. display/softkey format reference code
5. reference to the related Display/Softkey Format graphic
6. display/softkey format title
7. softkey type (e.g., momentary action, alternate action)
8. off-page transition designator

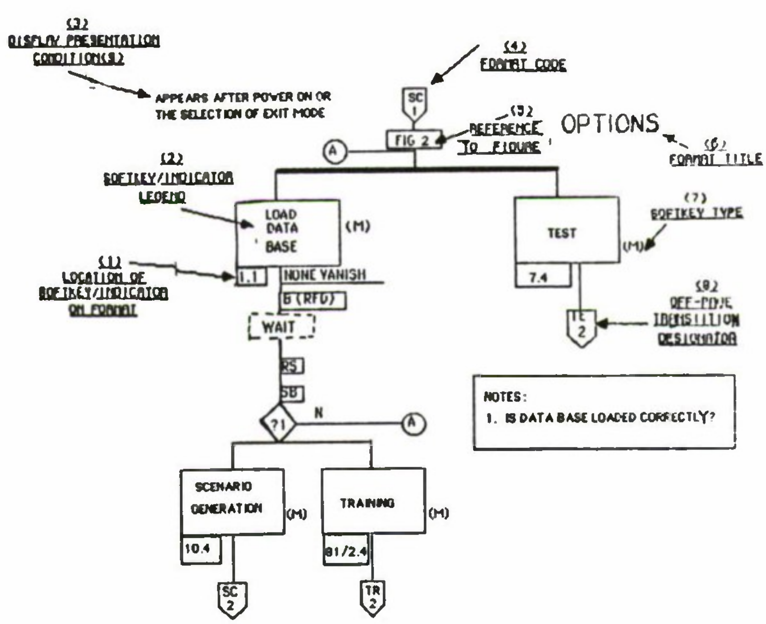


Figure 7. *Switch Control Tree (SCT)*

SUMMARY AND CONCLUSIONS

The role of HE in the design of an adaptive information interface for instructors who will use the TCC has been an extensive one. As required by MIL-H-46855, HE was involved during the concept formulation and design phases of activity, and continues to be involved throughout the fabrication, test and training phases as well. The major activities performed by HE for the trainer programs using the TCC included: conducting Functional Analysis to determine the instructor interface requirements, designing interactive Display/Softkey Formats to meet the identified instructor interface requirements, conducting Task Analysis to validate the adequacy of the designs and documenting the display/softkey control software performance requirements.

Because of the conditional nature of the adaptive information type of display, the HE and software tasks are complicated tremendously. To overcome this complexity, tools were developed or refined to generate documents that concisely communicate requirements to software development personnel and to the customer. These tools include: Display Format Graphics, Page Control Trees and Switch Control Trees.

Initial informal assessments of the interactive Display/Softkey Control Formats indicate that they do appear to simplify the instructor interface. The innovative display information presentation and feedback techniques (e.g., prompts, data range cues, windows, inverse video) and adaptive softkey control style used to minimize instructor training requirements and workload appear to achieve these goals. Additionally, preprogrammed scenarios, which require minimal involvement with the TCC during the conduct of training, should maximize instructor time available for trainee performance evaluation activities.

REFERENCES

1. DI-H-7051. Human Engineering Program Plan. 1 June 1979.

2. MIL-H-46855. Military Specification, Human Engineering Requirements For Military Systems, Equipment And Facilities. 31 January 1979.

3. Hiss, R. System Engineering Methodology For The Concept Formulation (CONFORM) And Contract Definition Phases Of The Ship Acquisition Process. Draft of Raytheon SSD Technical Report, Portsmouth, R.I. 10 April 1968.

4. Rizy, E.F. Multimode Minehunting Sonar Display Technology Study. Raytheon SSD Technical Report, Portsmouth, R.I. 3 May 1980.

5. MIL-STD-1472. Military Standard, Human Engineering Design Criteria For Military Systems, Equipment And Facilities. 2 May 1981.

6. Clifford, A., DeFanti, D., Kanarian, M., Manning, H., and Rizy, E. Revised Guidelines For The Development Of Switch Control Trees. Raytheon Company, Submarine Signal Division, Systems Operational and Performance Analysis Department. 10 October 1986.

ABOUT THE AUTHOR

Dr. Mary Kanarian is a Senior Scientist-Research employed by the System Performance And Simulation Analysis Department at Raytheon Company, Submarine Signal Division. She holds a Ph.D. in Experimental Psychology and has five years of HFE experience including: participating in research involving submarine command operations and decision analysis; research and design activities involving console design, computer display menu format design, interactive control panel design and training operations and procedure development. She has also completed HFE continuing education courses. Prior to her association with Raytheon and since, Dr. Kanarian has been actively engaged in research activities concerning various topics in Experimental, Developmental and Social Psychology. She also teaches courses in these topics at the college level.

TRAINING WITHOUT SCHOOLHOUSES

G. L. SCHILE
Chief of Naval Technical Training
Memphis, Tennessee

ABSTRACT

Technical training in the twenty-first century needs to adapt high technology to instructional methodology. Increased levels of technical skills will be taught in a climate of fewer dollars and with fewer active duty personnel available for instructor duty. This paper reports the results of a preliminary study to improve training in the twenty-first century in this climate.

Some of the alternatives explored include: contracting out entire training centers, life-cycle contractor training of weapons systems and/or selected equipments, and use of future information systems to reduce or eliminate the physical co-location of students and instructors. Areas for further study which have been identified include: identification of required information system capabilities; application of artificial intelligence to course design, development, and delivery; design of low cost generic terminals; and development of an algorithm which aids in the identification of factors essential to successful delivery of remote instruction.

INTRODUCTION

Futurists range from over-pessimistic doomsayers to those who foresee only "streets of gold" in a future Shangri-La. In fact there is a range of possible futures which we can influence by action taken or not taken today. However, rational thought based on reasoned judgment and past experience narrows the range of possibilities considerably. We need to make families of assumptions and determine the research and small scale pilots that need to be conducted now that will aid in making enlightened choices in the future. Two major premises undergird the alternatives explored in this paper. First, the number of technically trainable and recruitable young adults will decrease through the end of this century which will create a dearth of experienced military personnel available for assignment to instructional duties in the early twenty-first century. Second, increased complexity of new weapons systems and an ever increasing ability to change systems already in the fleet will increase the amount of training required, particularly for career personnel.

ALTERNATIVES

Three alternatives which can reduce the number of required military personnel assigned to training are:

1. Contracting out training centers.
2. Life cycle training of selected courses at contractor facilities.
3. Using high technology communications systems in support of selected training without formal schoolhouses.

Obviously each of these alternatives has advantages and disadvantages. The last two alternatives will require additional analysis and research to determine the limits of feasibility and parameters for optimum implementation.

Additionally, other alternatives also need to be explored such as shifting more of the front-end skills into public and private technical schools by such means as providing curriculum at no cost to these institutions and offering incentives to new assessions who have these skills.

CONTRACTED TRAINING CENTERS

The first alternative to compensate for lack of military instructors is to award a "turn-key" contract to run an entire training center. The Navy has contracted out selected maintenance and instruction at a number of training centerns since 1980. Currently over 1,000 contract personnel are teaching Navy courses in Navy run facilities. Depending on geographical area, course content, and experience of potential bidders, there appears to be a 10 to 20% life cycle savings of contractor personnel over a comparable military staff. This results from fewer turnovers, shorter average break-in time, and decreased personnel support requirements. Assuming a stable contractual work-force, costs of such incidentals as security clearances are actually lower compared to a military counterpart because of lower turnover rates.

It is the "military" component, not the "technical training" component of the training process, that will limit the applicability of contract instruction. Obviously recruit training requires "blue suit" examples of military standards of personal excellence. At the other extreme, training for career personnel should emphasize technical content and the choice between contractor or military instructional staff should be primarily economic in the broad sense, i.e., military shore duty billets that provide meaningful shore duty assignments to achieve acceptable sea-shore rotations is part of the economic equation. The value of the sea-shore part of the "equation" shifts the decision

toward contracting if future technology increases result in significant increases in length of training for career personnel.

Virtually every function of a training center could be contracted except for quality control, military models of personnel excellence, and where skills are not available in the private sector, e.g., some tactical skills.

This alternative to contract out an entire training center is a low risk option and can be implemented at any time. The Defense Department has extensive experience both in contracting parts of the training function (i.e., instructor contracts and maintenance support contracts) as well as "turn key" contracts such as Vance Air Force Base.

LIFE CYCLE TRAINING AT CONTRACTOR FACILITIES

If one were to compare the cost of training historically conducted at contractor facilities to the average cost of similar training in government facilities, the latter would show a much lower cost per graduate. However, that is due in part to the following:

1. Historically, contract training has been limited to one or two initial cycles of training. Therefore, course start-up costs tend to be spread over those few cycles of training. Economies of scale are not possible under these circumstances.

2. Where unused government facilities exist, the marginal cost of adding additional training appears much lower than expensive initial factory training which includes facility cost.

Additionally, because initial factory training is conducted with unproven curriculum, often with insufficient training equipment, and sometimes with instructors without instructional backgrounds, many people within Navy do not view contractor training as a long term alternative. One example of longer term contractor training is the recent contract for bridge training for officers at Newport.

There are a number of factors that mitigate against more of this type of training:

1. High cost of capital investment.

2. Short length of contracts (five years or less).

3. Government protection as a result of unsatisfactory performance (both monetary and ability to produce a continuous stream of graduates).

4. Extent of military presence needed (new assessions versus career personnel).

These factors are interactive in affecting decisions to opt for life cycle contracting. The higher the start-up costs, the longer the contract life needs to be in order to spread investment costs across the life of the contract. However, as length increases, particularly with capital intensive training equipment, government protection against less than optimum performance decreases, e.g., a contract for a 50 million dollar hot plant in the middle of XYZ Corporation, cannot reasonably be terminated since lead times for construction of replacement facilities would be several years. Laws are needed which allow expeditous judicial decisions to resolve conditions of unsatisfactory performance. The situation is not that different in nature than a ten or more year ship construction effort by a non-government shipyard and would need to be approached in much the same way.

Research

Additional research and analysis would be needed to establish optimum contracting procedures in order to begin life cycle contracting on a large scale. Military training managers, Navy comptroller personnel, contract specialist, and industry should be able to work out reasonable procedures and safeguards.

Another research implication is the development of an algorithm to assist in making decisions on what kind and how much military presence is required during technical training to develop and/or retain the purely military aspects of career development. To repeat a worn but nevertheless true cliche "a sailor first and a technician second". What is the trade-off between such factors as length of service, length of training, type of training, and the ability to "civilianize" the technical training component of personnel development?

A final research question regarding life cycle contracting concerns the size of the contracting effort. Should a contract cover a single course, a series of courses comprising a pipeline, several related pipelines, or a major portion of a warfare area. Economics of scale, synergistic effect of related training, and the sharing of common high value resources would tend to make one decide that large blocks of training should be contracted. Another aspect of size of life cycle contracts is the scope of indirect support and what is termed base operations support at military training centers. Full berthing, messing and recreational provisions could be specified in the contract allowing a wide latitude to achieve the end goals of such support. Civilian "mirror images" of traditional military training installations would be the easiest to define but may not be the best alternative, e.g., integration of training in a vocational-technical school setting (a variation of ROTC) may be a better approach.

Training contracts with full quality

of life support should not be written in traditional contract language of some specified number of man weeks of training or a given number of square feet of living space. Contract language should be developed to specify quantifiable skills or attitudes to be attained from an entry level baseline and some quality of life quotient to be maintained during the student's assignment for training.

TRAINING WITHOUT SCHOOLHOUSES

This option has the highest risk but greatest potential to increase quality of training and decrease cost of training of the alternatives explored herein. The concept should be very compatible with what many project as the information based society of the future. Any future implementation will require research and development in two general areas -- communications and instructional technology. Needed improvements in these two general areas will be discussed first, followed by a description of one possible twenty-first century scenario that reduces the need for schoolhouses at formal training centers.

Low cost communication is a prerequisite to make this alternative practical. An interactive network between two or more training stations would be needed without geographical constraint. Ideally, the interchange would include data, audio, video, and even holographic images. An optimistic view would be that such terminals would be in place in most homes, replacing existing home computers, video recording and playback equipment, tape and record decks, video games, libraries, television sets, and telephones. The effect would be more than replacement but the synergistic effect of totally integrating all of the present day capability. Such a terminal would be very sophisticated and complex yet simple to operate. Input-output modes would include voice, touch, pictures and text just to name a few. With such capabilities, the terminal or station would be capable of generic simulation of many future work stations. Holographic imagery would even create some part task training capability for psychomotor as well as the purely cognitive skills. One could conceivably be able to "touch" locations of analog controls and other physical components on holographic images.

Six instructional technology areas need to be enhanced from present day capability: artificial intelligence (AI) in curriculum development and instructional delivery, competency based evaluation systems, reduction in the amount of hands-on training required on operational equipment in formal training settings, electronic transportability of generic simulations, teleconferencing, and embedded training in operational fleet systems.

Artificial Intelligence

Today attention to format, cut and paste techniques, typing, art work, and other more mundane aspects of curriculum work consume disproportionate amounts of curriculum development resources. Artificially intelligent expert systems of the future, exercising control of future data bases, will enable subject matter experts and instructional experts to make major modifications and minor adjustments to curriculum very easily in almost real time.

One important application of AI is the adaptation of video gaming to instruction. A number of games currently exist that are used in formal Navy courses as well as to maintain and refine skills in the fleet. The Naval Personnel Research and Development Center has developed some gaming based training or assessment tools, e.g., Battle-Management Assessment System (BATMAN) and Raid Originator Bogie Ingress (ROBIN). Other wargames are being developed for specific applications. The long term critical need is to integrate AI based gaming expertise with subject matter expert AI so that these games can be produced and updated quickly and inexpensively.

On the instructional delivery side, yesterday's individualized instruction did not work well because in too many cases instruction consisted of reading paper texts assigned by an inflexible computer managed system that was supported by learning supervisors performing largely clerical functions. Continued growth in expert systems should allow subject matter expertise to be combined with instructor expertise in software of the future that would be cognitively comparable or even better than the average instructor today.

Competency Based Evaluation

Testing practices can be dramatically improved with future technology. Today multiple-choice questions and short answer types of objective tests are prevalent for testing material learned in the classroom. Some retesting of sub-areas on a test is currently practiced in some courses, while few Navy schools provide specific feedback in the form of remedial prescriptions. Performance testing needs to be improved to identify subtle knowledge or skill deficiencies. More complex methods of evaluating an individual's range of competence and assignment of finely tailored remedial instruction is too complex and time intensive today because instructors are needed to do the job; therefore, it is too costly. Again, artificial intelligence can be used to make complex comparisons and quickly generage easy to understand profiles of performance along with individually tailored prescriptions for remediation.

Reduced Hands-On Training

Reducing the amount of hands-on training is not likely to be well received by many trainers. However, there are several reasons why present day objections will be less valid in the future: jobs are becoming more cognitive and less psychomotor in nature, simulation techniques are improving, simulation is

easier because more jobs involve interaction with standard display terminals, and finally operational software which will have built-in training modes can be more easily transferred to simulators for part task training. The above factors will enable hands-on training to be concentrated near the end of the course or transferred on board ship.

Electronic Transportability

Computer based lessons will be more easily transportable over communications links. A particularly critical capability needed will be near real time interactive simulation at the various terminals.

Expert human instructors will need to be linked directly with one or more students. Today's teleconferencing and interactive television experience will provide a basis for future systems. Several systems are currently in use. One remote delivery system under test by a DOD training organization provides one way video and two way audio using compressed band-width techniques. Further progress is needed in integrating live camera with stored video. Low cost two way video is needed to further enhance present day systems. Instructor stations which present video from a "class" of students on a master display will recreate much of the present day classroom environment.

Embedded Training

Finally, increased embedded training capability in operational equipment will allow the shift of increased training to the fleet as well as reduced reliance on resident school hands-on training. Ideally the requirement for embedded training will be part of all new weapons systems procurement specifications. As mentioned earlier, subsets of fleet software can be used in generic terminals for individual training and even some team training. The same compatibility of training modes between weapons systems required on board will also allow subsets of training software to interact in a shore-based team training setting.

TRAINING SCENARIO

This scenario will combine various aspects of the three alternatives described above. The year is 20XX. The specific year XX is dependent upon the costs of communications/information systems and instructional technology available.

Recruit training will still need to be conducted by military instructors while most base support functions will be performed under contract. Initial assession general skills training, i.e., class "A" Schools, would vary from training at contractor facilities, contractors teaching at Navy facilities, to USN instructors at Navy facilities and combinations of the above. Computer assistance will be prevalent for remedial instruction, practice, and testing. Also by electronically sharing curriculum and generic simulation, some portion of many entry level schools will be taught in high schools and vocational-technical institutes.

The most radical changes should occur in equipment specific operator and maintenance training. Much of this training will be able to be structured so that the front-end can be supported by generic simulations on low cost terminals. This will result in front-end training being relatively site-independent, i.e., learning could occur on-board ship, at home, in civilian educational/technical institutions, at contractor facilities, or all of these. One typical example would be a young person being assigned to learn a new system via terminal on board ship. During his off-duty hours, he could, if desired, continue the lessons at hime when the ship is in port, or even at a local technical school. Learning the new system could be a temporary duty assignment for a number of weeks at one of these locations.

Training will be more continuous on a systematic basis versus the present prevalent method of front end loading training immediately after initial enlistment in residential schools. This is more flexible method of delivering instruction would allow immediate sea duty assignment after the initial class "A" School and then allow the sailor to continue training on-board through a video linked terminal. Depending on the instruction, the method could be purely computer-assisted (remember that extensive artificial intelligence provides both subject matter expertise as well as instructor expertise) or through communications links providing one-on-one tutorial with a real instructor or multistation interactive distributed quasi-classrooms, i.e., the instructor at one physical location and various class participants each at separate locations. If hands-on training cannot be accomplished on on-board systems either because of lack of time on equipment, unavailability of supervisory personnel to monitor the student, etc., then a short hands-on capstone segment of training would need to be conducted in formal training laboratories. Again, there would be a variety of ways to provide this capstone training, USN facilities run by the Navy, contractor run Navy facilities, or life-cycle contractor facilities.

Because software and hardware change to systems of the future will be increasingly easier to make, operator and maintenance skills must also change more rapidly. Lumping change into residential school modules and sending each operator and maintainer back for training several times a year is not practical now nor will it be in the future. Remotely delivered instruction is a way to keep the fleet up-to-date.

LONG TERM IMPLICATIONS

The long term implications of such

a scenario would be:

1. No dichotomy of training design and management between most sea and shore training.

2. Reduced military training facilities due to use of terminals in work spaces, homes, and civilian schools as well as increased use of contractor facilities. Berthing, messing and other support capability would be similarly affected.

3. Design of training would more readily accommodate reserve training.

4. Changes to personnel policy which would reward relevant training obtained prior to entry into the Navy as well as on off-duty time.

5. Tactics could be more dynamic because entire battlegroups could receive training in new techniques in almost real time.

NEAR TERM RESEARCH

It is most critical to recognize near term implications as to where research and development must be focused. These are some of the more obvious:

1. Improvements in artificial intelligence based curriculum development and delivery systems.

2. Development of sophisticated learning terminals; preferably for economic reasons these future terminals would be enhancements of common consumer information equipment.

3. Refinement of teaching and learning management techniques relative to remote delivery of instruction. Development of an algorithm which allows the training manager of the future to decide which training setting is most effective for a given training requirement.

4. Small scale tests on the various concepts, i.e., remote instruction, life-cycle contracting of training, and methodology for concentrating hands-on training near the end of the training sequence, etc.

5. Determination of the amount of military presence required at various stages of training in an individual's career. Obviously subject matter, student characteristics, and method of instruction will interact with yet unidentified factors. The initial socialization process of recruit training must necessarily be conducted in a closed military system. However, if follow-on training is conducted in non-military settings, superior performance and lifestyle of military personnel may serve to attract the best and brightest of non-military contemporaries for naval careers.

6. Refinement of ways to improve team attitudes and skills in light of less traditional classroom groups. The same communications links used to teach can establish teams that could be even less artificially created than a group who traditionally were assigned to begin training on the same day in a course.

CONCLUSIONS

There are many possible futures that we can help to create. The future that will actually occur in the year 20XX will depend in large part on the ideas and decisions made today. Besides the research needed in the areas described above, organizations must look to how they need to structure themselves for the future.

In the informational age, traditionally separate organizations will be in close cooperation. The organizations responsible for training at sea and training ashore must function as a single entity since where training is conducted will become less and less a function of location or setting. Policy and directives must become more and more forward looking as acceleration of change increases in the informational age in which the scenario described above occurs.

ABOUT THE AUTHOR

Dr. Gerald L. Schile currently heads the International Training Branch on the staff of the Chief of Naval Technical Training. Dr. Schile received his doctorate from Memphis State University. He also received the Secretary of the Navy Fellowship for Educational Specialists in 1981. His experience includes Training Program Coordinator for Consolidated Naval Electronic Warfare Training, Data Systems and Data Processing Technician Training, and Electrical Schools Training. Dr. Schile has taught and developed curriculum in a number of technical training schools.

THE LEARNING ARCADE

by William L. Maloy and Nancy N. Perry
Chief of Naval Education and Training
NAS Pensacola, FL 32508-5100

ABSTRACT

Rather than lament television watching, a few creative souls have exploited this medium to encourage young people to master instructional content. It is surprising that these successes, like Mr. Wizard and Sesame Street, have been widely acclaimed, but seldom replicated. It seems easier for many of us to criticize this media as a detractor from traditional learning rather than intervene to build effective and challenging TV-based learning models.

Similarly, video arcades are viewed as negative influences upon traditional learning systems. But, like the TV, these are opportunities in disguise. This paper focuses upon arcades and proposes that rather than lament their popularity, we exploit their applications to build desired cognitive and motor skills -- Learning Arcades, if you please.

Our discussion focuses on the Navy environment where there are opportunities for using previously untapped time for learning in locations not ordinarily thought of for this purpose. Furthermore, we suggest the learning arcade concept is applicable to the civilian community as well.

INTRODUCTION

In many highly structured schoolhouse environments, there are ample reasons for those in charge to resist packing more content into already overcrowded schedules. Added Navy training requirements, for example, are seldom accompanied at the schoolhouse level by a zero-based curriculum review. As a result, the schoolhouse policy maker is often faced with competing objectives: reducing training time on the one hand, while adding new training requirements on the other.

Since Navy training schedules are usually full, we need to explore opportunities to complement and build upon formal learning time. This is possible. For example, we have every reason to believe that we can take greater advantage of our young people's interest in video games. To do this we must develop captivating gaming software and place it in other than traditional learning settings which might prompt people to use their own time to learn. This, in essence, is the Learning Arcade concept.

The purpose of this article is to discuss in greater depth our plans for the establishment of a Navy Learning Arcade. We will also describe our initial plan for evaluating the effectiveness of the concept as well as point out management, instructional support and motivational training doctrine issues that must be addressed before a full-scale Learning Arcade can be established. Throughout, our discussion will also suggest the possibility of voluntary time as an avenue to enhance learning, and through the application of various gaming techniques to encourage its use for this purpose.

INITIAL DEVELOPMENT PLAN

To investigate the Learning Arcade concept in an orderly manner, the Chief of Naval Education and Training has decided to try out an arcade type of device as one approach that responds to a new requirement for Navywide firearms indoctrination to counter terrorism. We will place three M-14 Marksmanship trainers, designed by the Navy Training Systems Center, (NTSC, 1986) in preliminary Learning Arcade environments. Two will be in the barracks recreation area at an Apprentice Training site, a four week school that follows Recruit Training for those sailors going directly to the fleet. The third trainer will be placed in the wardroom of an NROTC Unit.

This marksmanship trainer consists of a demilitarized M-14 rifle connected to a light pen that is interfaced with a microcomputer. The light pen is aimed and fired at a target on the computer screen in the same way a real rifle is aimed and fired at a designated target. Weapon recoil and report are simulated to make the trainer more realistic. To provide further training applications, variation in distance, wind velocity, and target movement will also be added.

The system collects real time performance and physiological data which include breath rate, trigger squeeze pressure, rifle butt pressure, and weapon position. It executes a set of rules for analyzing these data and provides feedback to the trainee. In our initial evaluation, we are most interested in three things:

1) Will the Marksmanship Trainer provide for transferability? In short, will this trainer effectively and efficiently support Indoctrination level standards set for marksmanship throughout the Navy?

2) Will the trainer be durable enough to withstand heavy and constant use in nonsupervised situations?

3) Beyond indoctrination level transferability, how much can we substitute trainer practice for range practice? This information will serve as a basis for determining cost avoidance.

MANAGEMENT ISSUES

Regardless of our prime reason for setting up a successful learning arcade, it pays to spend a Friday evening in a video game room, observing the way it is organized and its activity level. As Turley points out, "Today's teenagers aren't pouring their quarters into those one-armed bandits in the shopping mall arcades because they are boring" (1985, p. 37). How does the arcade structure motivate so many of them to drop in one quarter after another from the time school gets out until the mall closes?

Even though our plan is not based upon the traditional profit motive, it is intended to provide a reasonable extension of formal instruction to increase readiness. Our goal, then, is the same as that of the profit based game room: to get sailors to play the games as much as possible. With this in mind, we have every reason to examine arcade management patterns and adopt what will work for us.

Site Supervision

The first thing one notices about a video arcade is how few adult employees are present -- usually one, and that person is hardly an instructor. Instead, his or her primary role is to monitor equipment: to see that the change machines keep dispensing quarters and to call a repairman if a game should stop working. Perhaps the adult could mediate disputes about who is entitled to what machine for how long, although the rules for that seem to be implicitly understood and followed without intervention. Finally, a "supervisory" presence may serve to deter vandalism.

The games themselves provide complete directions for their use and any discussion of strategies for improving one's playing of the game takes place among the players themselves. One suspects arcade attendants are not usually skilled at playing the different games, nor do they have any particular desire to gain these skills. Again, their focus is upon equipment performance rather than player performance.

What implication does this have for setting up the Learning Arcade? It does suggest that we must provide a notification plan if machines break and initially, at least, we must be alert to the possibility of deliberate or careless mistreatment of equipment. Beyond that, the emphasis should be upon games and content which promote serious, purposeful activity. In all probability we can get along with no more supervision than we presently find in most Navy living and working areas.

And how can we ensure that the games get used if there is no authority figure there telling people everybody must play three games before going to the mess hall, or everybody must reach level four by Friday? It appears that this can be accomplished best by making the directions for playing the game easy to follow and by creating the game so that it is inherently challenging and satisfying.

Finally, sharing time on the game doesn't appear to present a major problem. Visiting the typical Navy recreational center corroborates what we have already seen in the typical arcade environment: whether it be Pacman, pool, ping pong, or TV channels, people will civilly work things out among themselves if they are expected to do so.

Location

As we have noted, the primary goal of the arcade should be to make it easy for people to use their own time to pursue learning that interests them -- and that is of use to the Navy. There are a number of circumstances where voluntary time is potentially available; time which allows people to do with it what they will. Given interesting alternatives, most people would choose not to waste this voluntary time but apply it to learning tasks.

Prime targets are indoor recreational areas and such traditional "hurry up and wait" spaces as medical and dental waiting rooms. We will seek to create a low key environment with minimal rules and supervision rather than a formal schooling atmosphere.

We are mindful that the characteristics of certain games will make them wholly inappropriate for placement in a number of surroundings. Even moderate noise levels, for example, would prove disruptive outside most recreational areas.

Still, there are plenty of places where we can further test the concept if the pilot effort looks promising. Moreover, if this basic idea proves feasible we will also be alert to developing games which provide general information to the general Navy population

while furnishing more specific retraining games for the skilled technician in whose spaces the equipment is located. For example, games in medical areas could provide content ranging from emergency first aid procedures for common shipboard accidents to sophisticated protocols related specifically to the medical rating.

Citizenship

Finally, there is a concomitant issue to supervision and location matters. The Navy is giving a great deal of attention to ethical behavior in support of its goals for Pride, Professionalism and Personal Excellence. Fundamental to ethical behavior is good citizenship, with due regard for the rights and property of others. The placement of the Learning Arcade in generally unsupervised spaces would carry a clear message that the Navy expects its people to act as good citizens and that they should expect the same from each other.

In short, informal learning environments, that contain moderately expensive equipment will encourage us to raise our expectations about the way people behave. Our underlying belief is that if we expect them to -- people will act with due regard!

INSTRUCTIONAL ISSUES

To ensure that the learning arcade is instructionally effective to the maximum extent possible, we must ask two questions:

> How can we make the games so attractive people will play them?
>
> How can we make certain people will learn when they play?

In this section we attempt to answer these questions by describing motivational and learning strategies that can be designed into Learning Arcade games.

Motivation

Since use of the Learning Arcade will be voluntary, it is essential that the games be inherently appealing: those factors that make video arcade games fun to play should be deliberately selected and capitalized upon as the basis of motivational design strategy. We have listened to Myers' admonition that the difference between recreational and educational games is that the former appeal more to the players, while the latter appeal more to parents and teachers (1984, p. 182). We intend to appeal to the players: the central principle is to employ gaming strategies that will keep sailors involved.

In addition to our literature search, we interviewed several young Naval Officers to determine what arcade game features would allow us to adhere to this principle. Their reports suggest the magnetism of these games stems from the same strategies that learning psychologists would recommend for encouraging time-on-task for any instructional activity! Among these are: providing several skill levels to accommodate different levels of expertise, builtin scoring and bonus scoring gimmicks, allowing recognition of success, realistic graphics and sound effects, and allowing the number of people who can play at one time to vary.

Additionally, we believe the design of these games must relate Navy content to Navy requirements in a manner that will appeal to a Navy member's sense of pride, professionalism, and personal accomplishment. Simply designing games around Navy scenarios is not likely to be enough.

Multiple Skill Levels. Multiple skill levels are motivating because they keep the game challenging. A game is fun to play only as long as it is neither too easy nor too difficult: the same six year old who disdains Candyland may be frustrated to the point of tears by Choplifter. The implication for the Navy is that the optimal use of the same game would result from a design which permits skill development throughout a continuum of proficiency.

Scoring Schemes. Most arcade games have intricate scoring schemes, allowing points for accomplishing the basic task, extra points or other rewards for reaching certain designated levels in the game, and intermittent opportunities to earn other bonuses. Building these schemes into a game increases its complexity and greatly enhances its challenge, thereby sustaining the interest of the players for longer periods of time.

Recognition of Success. Another feature of arcade games that our interviewees liked was an opportunity for high scorers to enter their initials into the game program so that they were displayed for all to see. In a Navy recreational area, as in a video arcade, this extra bit of recognition might well prove inspiring to those who like to be known as the best.

Realistic Graphics and Sound Effects. Gone are the days of Pong. Low resolution graphics and simplistic sound effects no longer attract video game players. The more a game simulates the visual and aural components of a real world situation, the more willing video game aficionados are to play.

Variation on the Number of Players. To broaden the appeal of a game, provision should be made for allowing someone to play alone for self improvement as well as providing two or more competitors the

around which technology based games can be built to enhance individual and group performance by encouraging voluntary learning in a Navy Learning Arcade setting. They range from management techniques to game formats, and they require that we take full advantage of what we already know about motivational strategies and learning principles.

The pilot project should enable us to think more broadly about optimal Navy applications, should help us realistically address concept and operational limitations, and should allow us to articulate more clearly this program's potential for our nation's civilian community. These are broader issues we intend to address and report as part of the pilot evaluation process.

CONCLUSION

Across all levels of the military training establishment, policy makers and schoolhouse managers are calling for the imaginative use of technology in support of the teaching and learning process. The civilian education and training community has likewise discovered creative techniques for increasing learning with computers. In both settings instructional software is beginning to more fully exploit the hardware's expanding capablity. We join with those who see great potential in the use of technology to enhance human performance, and this paper has suggested ways in which that can be done through gaming applications. But we go beyond merely endorsing the applications of technology in support of better classroom instruction by suggesting it has a major role to play in helping us design learning opportunities outside traditional instructional settings.

We have noted that in our Navy world time and money constraints make it difficult to add more and more required instruction to already crowded schoolhouse schedules. This makes it imperative that we pay attention to the fact that learning is, indeed, taking place in many diverse settings. Moreover, in a number of these settings, many people are highly motivated to learn on their own; a second fact that seems to escape those of us preoccupied with formal classroom training alone.

We argue that we should begin to identify skills and knowledge, the teaching of which could be deferred at initial formal Navy training sites, and, instead, be presented when and where people could apply voluntary learning time to acquire them. If this can be done in a Navy environment, the Learning Arcade concept can serve as a model for a sustained learning continuum with which to address the lifelong learning requirements of 21st century America.

BIBLIOGRAPHY

Gagne,' R.M. and Briggs, L.J. Principles of Instructional Design. New York: Holt, Rinehart, and Winston, 1974.

Gagne', R.M. The Conditions of Learning, 4th ed. New York: Holt, Rinehart, and Winston, 1985.

Hayes, W.A. Skill and Knowledge Assessment Tool (SKA^T): Test Evaluation and Report. Unpublished Chief of Naval Education and Training Paper, 1986.

Myers, David. The patterns of player-game relationships: a study of computer game players. Simulation and Games. 15 (2), 1984, 159-85.

_____. 1986 Project Profiles. San Diego: Navy Personnel Research and Development Center, 1986.

_____. Research and Development Department Project Summaries. Orlando, FL: Naval Training Systems Center, 1986.

Peters, R. DeV. Computer-Based Education and Training Functions: A Summary. (NPRDC Technical Note 82-17) San Diego: Navy Personnel Research and Development Center, 1982.

Silvern, S. B., Williamson, P.A., & Countermine, T.A. Aggression in young children and video game play. Paper presented at the biennial meeting of the Society for Research in Child Development, Detroit, MI, April 22, 1983a.

Silvern, S.B., Williamson, P.A., & Countermine, T.A. Video game play and social behavior: Preliminary findings. Paper presented at the International Conference on Play and Play Environments, Austin, TX, June 29, 1983b.

Turley, Gerald H. Exploiting "Arcade Technology" for Increased Readiness. Government Executive, 17, 1985, 37-40.

Wisher, R.A. The Development and Test of a Handheld Computerized Training Aid. Unpublished Army Research Institute concept paper, 1985.

ABOUT THE AUTHORS

DR. WILLIAM L. MALOY is the Director of Education and Training Development on the Staff of the Chief of Naval Education and Training, Naval Air Station Pensacola, Florida. He has oversight of a variety of programs including: NJROTC; Professional Development of Officer and Enlisted Personnel, including Off-Duty Voluntary Education; Training Technology Implementation; and Curriculum Development. Dr. Maloy holds B.A. and M.S. degrees from the University of Nebraska, his Doctoral work was in Elementary, Secondary, and Teacher Education at Florida State University in

1958.

DR. NANCY N. PERRY is Head of Training Technology Implementation on the staff of the Chief of Naval Education and Training, Naval Air Station Pensacola, Florida. She is currently responsible for CNET's involvement in: the Navy's Skill Enhancement Programs in support of the Chief of Naval Operations' Personal Excellence Program including Functional Reading, Functional Math, and Computer Literacy; On-Board Training; and the Model School Program. Dr. Perry holds a B.A. in Chemistry from Michigan State University; a M.A. in Instructional Development and Technology from Michigan State University and an Ed.D. in Instructional Design and Development from Florida State University.